ADAPTIVE CRITIC DESIGNS BASED NEUROCONTROLLERS FOR LOCAL AND WIDE AREA CONTROL OF A MULTIMACHINE POWER SYSTEM WITH A STATIC COMPENSATOR

A Dissertation
Presented to
The Academic Faculty

By

Salman Mohagheghi

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
August 2006

ADAPTIVE CRITIC DESIGNS BASED NEUROCONTROLLERS FOR LOCAL AND WIDE AREA CONTROL OF A MULTIMACHINE POWER SYSTEM WITH A STATIC COMPENSATOR

Approved by:

Dr. Ronald G. Harley, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. David Taylor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Deepak Divan
School of Electrical and Computer Engineering
Georgia Institute of Technology

Dr. Ganesh K. Venayagamoorthy
Department of Electrical and Computer Engineering
University of Missouri- Rolla

Dr. Thomas G. Habetler
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: June 13, 2006

To my grandfather, Mehdi Bazargan,
who was always a big inspiration for me…

ACKNOWLEDGEMENT

At the age of eight, I faced the first scientific challenge of my life, when I had to divide an even number by an odd integer. I thought if I learn how to solve that, I need not worry about anything anymore. Ironically, that was true…

This was not a journey one could finish on one's own. Many people have come into my life, and helped and encouraged me during the past few years. Acknowledging their endless support here is the least I can do.

I would like to thank my PhD advisor, Professor Ronald Harley, who constantly supported me during this path. His vision on the problems and how to tackle them, taught me how to approach the toughest challenges in my research.

I owe a lot to Professor Ganesh Venayagamoorthy for his invaluable guidance, help and technical advices throughout my PhD studies.

Also, I would like to thank my PhD proposal committee, Professor Deepak Divan, Professor Thomas Habetler and Professor David Taylor for their insightful suggestions and comments which helped me shape my PhD research in a more realistic way.

During my studies at Georgia Tech, I made many friends. I would like to thank them all for all the great moments I had with them: my friends and colleagues in the Power group, especially Satish Rajagopalan, Vinod Rajasekaran, Giorgos Stefopoulos and Yamille del Valle, and all my friends at the Iranian Student Association at Georgia Tech, my Iranian family in Atlanta.

When I look back at my accomplishments in life, I can see a clear trace of my family's concerns and devotion everywhere. My dearest mother, Mrs. Fattaneh Bazargan,

whom I owe everything I have achieved and whatever I have become; my beloved father, Mr. Mohsen Mohagheghi, for always believing in me and inspiring me to dream big even at the toughest moments of my life; and my only sister, Mona, who was always my silent support during all the hardships of this endeavor and beyond.

Last but not the least, to my dear Neda, who came along with me in every step of this journey and was always there for me whenever I needed her most.

I’m glad I started this dream, I feel proud for holding on to it and thankful for all the help, thoughts and blessings.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACD	Adaptive Critic Designs
ADACD	Action Dependent Adaptive Critic Designs
ADHDP	Action Dependent Heuristic Dynamic Programming
ADP	Approximate Dynamic Programming
ANN	Artificial Neural Network
ASVC	Advanced Static Var Compensator
AVR	Automatic Voltage Regulator
BESS	Battery Energy Storage System
COA	Center of Area
COEM	Controller Output Error Method
CSI	Current Source Inverter
FACTS	Flexible AC Transmission System
FC	Fixed Capacitor
FNL	Functional Link
GTO	Gate Turn-off Thyristor
HDP	Heuristic Dynamic Programming
LMS	Least-Mean-Square
MLP	Multiple Layer Perceptron
MOM	Mean of Maximum
NARMA	Nonlinear Autoregressive Moving Average
NTP	Network Topology Processor
PCC	Point of Common Coupling
PID	Proportional-Integrator-Derivative
PLL	Phase Locked Loop
PMU	Phasor Measurement Unit
POCS	Projection onto Convex Sets
PRBS	Pseudorandom Binary Signal
PSS	Power System Stabilizer

PWM	Pulse Width Modulation
RBF	Radial Basis Function
RTDS®	Real-Time Digital Simulator
SCADA	Supervisory Control and Data Acquisition
SSMF	Shrinking Span Membership Function
STATCOM	Static Compensator
STATCON	Static Condenser
SVC	Static Var Compensator
TCR	Thyristor Controlled Reactor
TCSC	Thyristor Controlled Series Compensator
TCT	Thyristor Controlled Transformer
TDL	Tapped Delay Line
TS	Takagi-Sugeno
TSC	Thyristor Switched Capacitor
TSR	Thyristor Switched Reactor
ULTC	Under Load Tap Changing
VCDim	Vapnik-Chervonenkis Dimension
VSI	Voltage Source Inverter
WAC	Wide Area Control(ler)
WASP	Wide Area State Predictor

SUMMARY

Modern power systems operate much closer to their stability limits than before. With the introduction of highly sensitive industrial and residential loads, the loss of system stability becomes increasingly costly. Reinforcing the power grid by installing additional transmission lines, creating more complicated meshed networks and increasing the voltage level are among the effective, yet expensive solutions. An alternative approach is to improve the performance of the existing power system components by incorporating more intelligent control techniques.

This can be achieved in two ways: introducing intelligent local controllers for the existing components in the power network in order to employ their utmost capabilities, and implementing global intelligent schemes for optimizing the performance of multiple local controllers based on an objective function associated with the overall performance of the power system. Both these aspects are investigated in this thesis.

In the first section, artificial neural networks are adopted for designing an optimal nonlinear controller for a static compensator (STATCOM) connected to a multimachine power system. The neurocontroller implementation is based on the adaptive critic designs (ACD) technique and provides an optimal control policy over the infinite horizon time of the problem. The ACD based neurocontroller outperforms a conventional controller both in terms of improving the power system dynamic stability and reducing the control effort required.

The second section investigates the further improvement of the power system behavior by introducing an ACD based neurocontroller for hierarchical control of a

multimachine power system. The proposed wide area controller improves the power system dynamic stability by generating optimal control signals as auxiliary reference signals for the synchronous generators' automatic voltage regulators and the STATCOM line voltage controller. This multilevel hierarchical control scheme forces the different controllers throughout the power system to optimally respond to any fault or disturbance by reducing a predefined cost function associated with the power system performance.

CHAPTER 1

INTRODUCTION AND OBJECTIVE OF RESEARCH

1.1. Problem Statement

1.1.1. STATCOM Internal Controllers: Challenges and Solutions

It is widely known that exchanging active and reactive power with a power system can help improve its steady state and transient performances. This can be achieved by inserting mechanically switched capacitors and reactors into the system, or applying power electronics based converter technology. The latter can provide a fast and continuous control over a wide range of operating conditions. This technology is referred to as flexible ac transmission systems (FACTS) (Hingorani and Gyugyi, 1999).

Essentially, FACTS devices can be classified into different categories based on the way they are connected to the power network: shunt, series and shunt-series. Shunt FACTS devices are connected to the power network in parallel and their main objective is to control the voltage at the point of common coupling (PCC). Series FACTS devices are inserted into the system in series with the transmission lines and are capable of regulating the active and reactive power flow through the line. Shunt-series devices are a combination of both shunt devices and series devices and are capable of functioning as both. Although there is a main control objective associated with each of the FACTS device categories, adding auxiliary control signals to each device can widen their control capabilities. Table 1.1 summarizes the properties of the basic FACTS device categories in various applications in power systems (Povh, 2000; Hingorani and Gyugyi, 1999).

Reactive shunt compensation is an appropriate method for regulating the voltage at a certain bus (node) in the network and to provide voltage support during small scale and large scale disturbances where the uncompensated network is not capable of restoring the voltage at the pre-fault desired level.

Table 1.1. Comparison of different FACTS device categories.

Application	FACTS Device Category		
	Shunt	Series	Shunt-Series
Voltage Control	High	Low	High
Load Flow Control (Meshed System)	Low/Medium	Medium/High	High
Transient Stability	Medium	High	High
Oscillation Damping (Interconnection Links)	Medium/High	High	High
Oscillation Damping (Meshed System)	Low/Medium	Medium	High

This study focuses on the applications of a specific type of shunt FACTS device: a static compensator (STATCOM), also known as static condenser (STATCOM) and advanced static Var compensator (ASVC). This device is connected in parallel to a power network and is mainly used for controlling the voltage at the point of connection to the power system.

Analytical approaches have been predominantly used for modeling and controlling power system components, such as FACTS devices. However, these mathematical models/equations are determined under certain restrictive assumptions, such as

linearizing a nonlinear system and/or approximating a higher order system by a low order model. Even under such conditions the solution will not necessarily be trivial, and sometimes uncertainties associated with real life problems further exacerbate the reliability of such approaches.

Many researchers have focused on the linear models for the STATCOM and the power network (Dong *et al.*, 2004; Wang, 1999; Wang and Swift, 1998; Schuader and Mehta, 1993), in which the system equations are linearized around an operating point. Based on these linearized models, conventional proportional-integral (PI) controllers are fine tuned in order to have an acceptable performance during system perturbations. While these models are appropriate for certain small signal applications in the vicinity of that specific steady state operating point, they cannot reflect the true nature of the power network and the FACTS device when the system is exposed to large scale faults or dynamic disturbances that change the configuration of the whole plant.

Moreover, the majority of the techniques in the literature that are used for fine tuning a PI controller are somehow dependent on a mathematical model of the plant to be controlled (Krajewski *et al.*, 2004; Clarke, 2003; Kristiansson and Lennartson, 2002; Suchomski, 2001; Poulin and Pomerleau, 1996), which is often only approximate in any case. In that case, such a fine tuned PI controller is not even optimal at that specific operating point. In fact, trial and error techniques are the most common method used for tuning PI controllers in industrial applications.

The disadvantages of the linear controllers/models, such as being dependent on the operating condition, can be overcome by using appropriate nonlinear control techniques. The well-established theory of nonlinear controllers range from fixed structure controllers

to more complicated adaptive, robust and optimal designs (Khalil, 2002; Bertsekas, 2000). While these designs can give good control capability over a wide range of operating conditions, they have a more sophisticated structure and are more difficult to implement compared to the linear controllers. In addition, they also need a mathematical model of the system to be controlled (Lu *et al.*, 2001; Liu *et al.*, 2003; Yao *et al.*, 1998). However, developing an analytical model for a power system becomes less practical for a large real world power network with numerous interconnections.

Artificial neural networks and fuzzy systems can offer solutions to the above problems. These are intelligent based techniques that can model/control a nonlinear non-stationary plant in the presence of noise and uncertainties. With proper training, intelligent controllers can provide nonlinear optimal or adaptive control, with no need to a mathematical model of the plant to be controlled. This advantage distinguishes them from all the analytical nonlinear control techniques.

1.1.2. Local vs. Global Control in Power Systems

An electric power grid in general consists of components such as synchronous generators, transmission lines, transformers, loads, active/reactive compensators, switches and relays. The compensators are shunt or series elements such as capacitors and inductors or power electronic converter based FACTS devices. The mathematical model of such a power network usually consists of thousands of states and multiple controllers with their own actuators and measurements.

With the power industry moving towards deregulation, the long distance power transfers between different regions –which can be hundred of miles away from one another- are continuously increasing. Moreover, with the introduction of more

industrial/residential loads, the system normally operates much closer to its security limits than before. Economical and sometimes environmental concerns often discourage the addition of new transmission facilities, which in turn could weaken the system against transient and/or dynamic disturbances.

Decentralized Control Structure

Typically, the voltages at the terminals of the synchronous generators are controlled by automatic voltage regulators (AVR) in order to maintain a proper voltage profile throughout the network. Until now, designs of the internal controllers of a generator (voltage regulator and governor) have traditionally considered only the single generator and ignored other controlled devices in the power network. Extra stabilizing capabilities are sometimes provided for certain generators by utilizing power system stabilizers (PSS), which are mostly designed to increase the damping of the local low frequency oscillation mode of the generator (Kundur, 1994). Similarly, designers of FACTS device controllers consider only the FACTS device and ignore neighboring generator controllers for example (Hingorani and Gyugyi, 1999).

In this *decentralized control* structure, each internal/local controller acts as an *agent* to minimize the local deviations or controlled variables, or maximize a local performance index without any information about the overall system objectives. The information and control responsibilities are distributed among these local agents of the system. Based on their own information, these agents make control decisions in order to comply with the desired behavior of the decentralized system (Yoo and Lafortune, 2004). It basically formulates the problem as systems whose specifications are given as global constraints but whose solutions are described by local controllers (Rudie and Wonham, 1992). By

adopting the concept of decentralized control, the system is controlled using local control stations; each is connected with a subsystem and receives only information concerning the local state variables. Nevertheless, all these controllers contribute to controlling the overall interconnected system (Bahnasawi *et al.*, 1990).

Therefore, all these internal control schemes, whether for the synchronous generator or the FACTS device, focus on controlling each component from an internal point of view, i.e., providing appropriate signals for the device in order to control some local quantity such as voltage, speed deviations or line power flow. However, with a number of these controlled devices close to one another in a power network, the issue of interaction between them arises, that at times can lead to adverse effects causing inappropriate control effort by different agents. This happens since each one attempts to be a good local controller, but has no information on the overall control objective of the entire system.

Moreover, modern power networks are affected by steadily increasing incidents of faults/disturbances that lead to inter-area rotor angle and power oscillations. As opposed to the local oscillation modes that are largely determined and influenced by the local area states, the inter-area modes are more difficult to study since they require a detailed study of the system as a whole and are influenced by the global states of larger areas of the power network. A locally tuned and designed PSS normally fails to perform effectively during the inter-area oscillations. A carefully tuned PSS may also be effective in damping inter-area modes up to a certain transmission loading. The effectiveness in damping inter-area modes is limited because inter-area modes are not as highly controllable and observable in the generator's local signals as the local modes are (Klein *et al.*, 1992). Also, the number of dominant modes in the system is much larger than the number of the

controlled devices available (Chow *et al.*, 2000). Therefore, the traditional approach of having a PSS tuned for its local mode, as well as a certain inter-area mode, might not be sufficient for ensuring the stability of the system.

Centralized Control Structure

The above issues raise the need for an alternative solution that can either replace the current local control system, or to work in parallel with the local controllers and help improve the system performance. A *centralized control structure* can be one solution. This technique controls the network using a single supervisory level controller which sends out all the required control signals to the various controllable components throughout the network. The first downside of such a scheme is that it cannot perform as fast as the local controller; therefore, control speed is the first tradeoff. Also, the price of failure of the main controller for such a scheme is very high; hence, it cannot be an appropriate solution for the power grid.

Hierarchical Control Structure

Another solution can be *multilevel control techniques*, also referred to as a *multi-agent hierarchical control structure*, a supervisory level control or wide area control (WAC). The objective here is to define a set of sub-problems that can be considered independent at a certain level (subsystem level). Through the manipulation of the interplaying effect at a higher level (coordinator), the global solution is obtained (Bahnasawi *et al.*, 1990). Different from both centralized and decentralized control, a multi-agent control structure employs a number of semi-autonomous agents that collaborate with each other to achieve a given task (Ni *et al.*, 2002). Such a regime requires communication and coordination not among all the agents, but only among those

closely related agents with common interests (Wang, 2001). Although the agents communicate with one another, each agent performs primarily based on its own interest; therefore care should be taken that no agent's actions should violate its own limits.

In an interconnected power network, the adverse interaction between agents could be avoided by using such a multilevel coordinated (supervisory) control strategy or WAC, which has prior knowledge of each agent's effect on the network. It is normally assumed that the WAC coordinates the actions of the various agents throughout the network by using the supervisory control and data acquisition (SCADA) system, phasor measurement units (PMU) or other wide area dynamic information systems (Karlsson *et al.*, 2004). The WAC would receive data from the power system and, based on the defined objective functions, would send appropriate control signals to the agents in the power network, in order to optimize the overall system performance as illustrated in Figure 1.1.

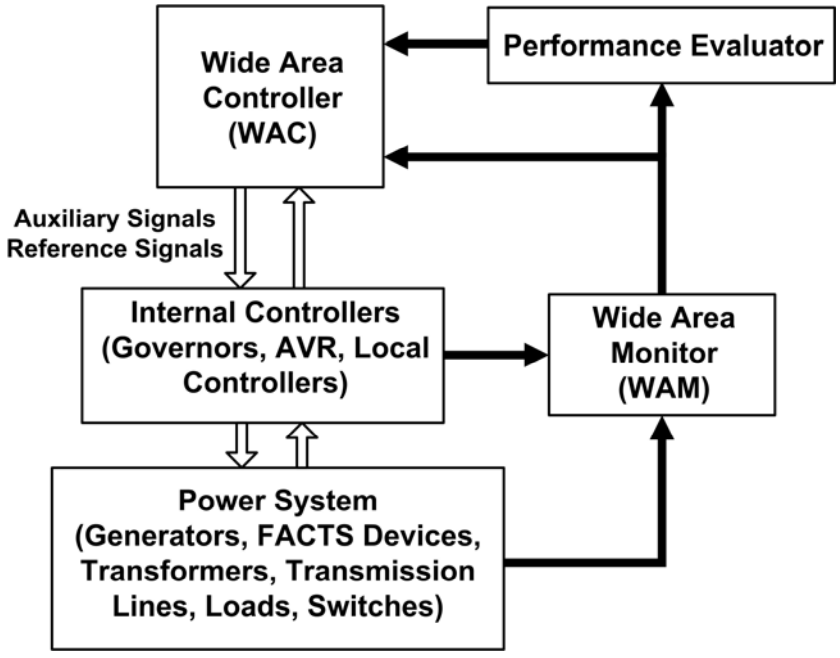

Figure 1.1. Schematic diagram of the WAC.

Several researchers have proposed linear or nonlinear controllers for providing hierarchical control in the power system (Ni *et al.*, 2002; Kamwa *et al.*, 2001; Chow *et al.*, 2000; Aboul-Ela *et al.*, 1996). Similar to the local controllers, these supervisory level controllers suffer from the disadvantages of being dependent on the operating conditions of the power system (for linear schemes) or requiring a mathematical model of the power network (for nonlinear designs). Once again, intelligent control schemes can be used as alternatives to the analytical approaches. Since a lot of the intelligent schemes are designed based on reinforcement learning, the size of the power system is not as crucial as it is in the case of the linear/nonlinear controllers.

1.2. Intelligent Control: A Reality?

Intelligent control techniques are the next generation of the nonlinear controllers that have the advantage of further improving the controller's performance by incorporating heuristics and expert knowledge into the nonlinear design techniques. While these intelligent designs have major advantages over the linear and nonlinear schemes, there is one thing common among them: their structures are considerably more complicated than that of a regular proportional-integral-derivative (PID) controller and are usually more expensive to implement.

More importantly, for the average engineer, the conventional PID is still considerably more understandable, easier to analyze and more of a white box compared to the intelligent controllers. Therefore, there is a tendency in practical power system applications to continue employing conventional linear and rarely nonlinear controllers; and intelligent controllers need more time to prove their efficiency and superiority over the traditional approaches.

However, the first step for incorporating intelligent techniques in power system control can be designing an intelligent controller for higher level (supervisory) control of the components in a power network. This way, an intelligent controller can be applied to auto-tune the conventional local PID controllers, provide auxiliary inputs for them, optimally adjust their set-points and suchlike.

Essentially, such an approach is similar to the nervous system in the human body where the higher level system (the brain) sends signals to control the set-points of simple nerve cells in the muscles, which are very similar to PID systems (Werbos, 1999). The brain acts as the central nervous system by performing the main decision making. The peripheral nervous system is constituted of local systems (muscles) that follow the command they receive from the brain (Figure 1.2).

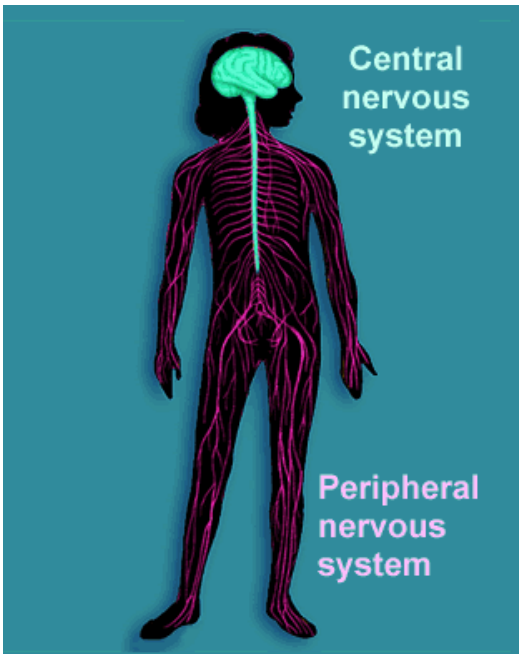

Figure 1.2. Hierarchical control in the human body nervous system.
Picture Courtesy of http://www.morphonix.com.

1.3. Objective of Research

The objective of this research can be classified into two parts:

The first part of the thesis focuses on applying intelligent control techniques for designing *local* adaptive/optimal nonlinear controllers for a STATCOM connected to a multimachine power system. These techniques do not need any mathematical model of the plant to be controlled and therefore, have a major advantage over the analytical nonlinear controllers. In this part, the main objective of the STATCOM is to control the voltage at the point of connection to the network and the proposed controllers deal with the local quantities/measurements only. The main goal here is to design intelligent controllers that not only provide faster and more accurate performance in controlling the voltage (higher efficiency), but they achieve this with less control effort generated (reduced cost). The procedure followed can be summarized as below:

- Designing a neural network based identifier (neuroidentifier) for small signal and large signal modeling of a STATCOM in a multimachine power system. Such a neuroidentifier is independent of any mathematical equation or model of the power system or the STATCOM, does not need any prior knowledge of the system and can be effectively trained to learn the dynamics of this nonlinear non-stationary system. The neuroidentifier can serve as a model of the power system in any intelligent control design,
- Design and hardware implementation of a fuzzy logic based controller for the STATCOM. Such a controller, as a nonlinear gain scheduling controller, can provide an acceptable performance over a wide range of the operating conditions of the power system,

- Designing an indirect adaptive based fuzzy controller for the STATCOM. Using this adaptive scheme, the performance of the fuzzy controller is further improved. A neuroidentifier can be used as the model of the power system, which is required for obtaining the correct update signals for the parameters of the adaptive controller,
- Designing an adaptive critic design based neurocontroller for the STATCOM in a 9-bus and a 45-bus multimachine power system, as a section of the Brazilian power network. Such a controller can provide optimal control over the infinite horizon of the problem in the presence of noise and uncertainties.

The second part of the research focuses on designing *global* (wide area) state prediction systems and controllers for a multimachine power system with a STATCOM connected to it. The complexity of the multimachine power system and the nonlinearities associated with it make it extremely difficult to design an analytical linear or nonlinear supervisory level controller.

Without loss of generality and for simplicity, the local STATCOM controller is considered to be a conventional PI controller in this section. The step by step procedure is as follows:

- Introducing a robust wide area state prediction system for a 12-bus multimachine power system with a STATCOM. Such a predictor, developed by neural networks, does not need any analytical information on the power system and its performance. Simply by observing the various measurements/states from across the power system, it is able to predict the values of the power system states at multiple steps ahead. By using a second neural network based structure, the state

predictor can be made robust to the partial loss of inputs due to failed sensors or transport lags. Such a wide area state predictor is essential to any wide area (global) control scheme for the power system,

- Designing a global neuro-fuzzy external optimal controller for a STATCOM in the IEEE 12-bus benchmark multimachine power system, using remote signals, in order to increase the damping of the system,
- Designing a global (wide area) optimal neurocontroller for the IEEE 12-bus benchmark multimachine power system with a STATCOM. The controller can be trained to provide auxiliary control signals to the different components in the network, such as the STATCOM or the synchronous generators, thereby help to improve the dynamic stability of the power system.

1.4. Thesis Outline

A brief introduction and the objectives of this research are described in this chapter. The rest of the thesis is presented in eight more chapters in which technical discussions and simulation/experimental results are provided, and a final chapter that summarizes the concluding remarks of the research as well as the suggested future work.

Chapter 2 gives a detailed explanation on the STATCOM, its structure and the conventional control schemes. Also, a conventional PI controller is presented for the STATCOM which is considered as the benchmark controller for comparison with the proposed intelligent controllers.

Chapter 3 presents a literature survey of the previous research carried out on designing internal and external controllers for the STATCOM as well as wide area

(global) controllers for multimachine power systems with synchronous generators and/or FACTS devices.

Chapters 4, 5 and 6 focus on local controller design for a STATCOM.

In Chapter 4, a neural network based identifier is designed for modeling a STATCOM in a 9-bus multimachine power system. A radial basis function (RBF) neural network is employed that acts as a global identifier of the power system.

Chapter 5 proposes two fuzzy logic based controllers for the STATCOM in the 9-bus multimachine power system: a fixed structure Takagi-Sugeno fuzzy controller as well as an indirect adaptive fuzzy controller. The adaptive controller is linked to a neuroidentifier as the model of the plant to be controlled. The former controller is also implemented in hardware using a real time digital simulator (RTDS®). Both the proposed controllers are shown to be more efficient than a fine tuned PI controller.

Chapter 6 extends the work in Chapter 5 by designing an adaptive critic designs (ACD) based optimal neurocontroller for the STATCOM. The performance of the proposed controller is evaluated in two systems: a 9-bus multimachine power system and a 45-bus 10-generator system as a section of the Brazilian power network.

The next three chapters consider the problem of wide area control of a power system.

Chapter 7 introduces a robust wide area state prediction system that can be used for predicting the parameters of a multimachine power system in the presence of a static or dynamic transport lag.

A neuro-fuzzy optimal external (supervisory level) controller is designed in Chapter 8 for a STATCOM in the IEEE 12-bus benchmark system. Using the remote signals in the

network, the proposed controller makes it possible for the STATCOM to improve the dynamic damping of the system.

Chapter 9 proposes an optimal wide area control scheme for the IEEE 12-bus system with the STATCOM. This wide area controller helps control the set-points of the different agents in the system (generator AVR, STATCOM line voltage controller) in order to improve the transient performance of the system.

Finally, conclusions, contributions and recommendations for future work appear in Chapter 10.

CHAPTER 2

STATIC COMPENSATOR- MODELING AND CONTROL

2.1. Introduction

A static compensator (STATCOM) is a shunt FACTS device which is mostly employed for controlling the voltage at the point of connection to the network. This chapter provides an introduction to the STATCOM along with a comparison with a static Var compensator (SVC), its structure and its applications in the power network. Modeling and control of a STATCOM are also explained in detail in the following sections.

In addition, a decoupled control structure is presented for the STATCOM which serves as the benchmark conventional controller in this study. All the intelligent controllers designed for the STATCOM in the next chapters are compared with the performance of this benchmark controller.

In this study, the STATCOM is modeled in the PSCAD/EMTDC® environment (see Appendix A for details).

2.2. Controlled Reactive Shunt Compensation

Shunt FACTS devices are connected in parallel with power system and have the ability to control the voltage at the point of common coupling (PCC). The SVC and the STATCOM are the most common devices of this type. An overview on the structure and performance of these two FACTS devices is presented in this section. More detailed explanations can be found in (Mathur and Varma, 2002; Hingorani and Gyugyi, 1999).

2.2.1. Static Var Compensator (SVC)

The schematic diagram of a SVC is shown in Figure 2.1. The SVC may include several different elements, namely, a thyristor controlled reactor (TCR), a fixed capacitor (FC), a thyristor switched capacitor (TSC), a thyristor controlled transformer (TCT) and a thyristor switched reactor (TSR); the first three elements are the most common ones (Hingorani and Gyugyi, 1999).

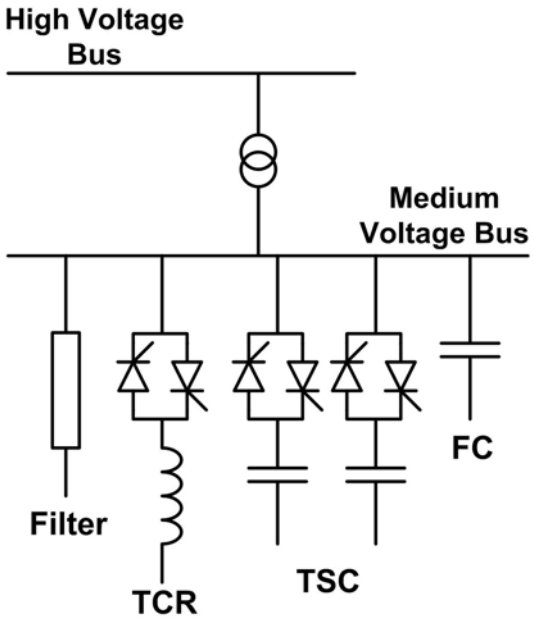

Figure 2.1. Schematic diagram of a SVC.

2.2.1.1. Thyristor Controlled Reactor (TCR)

TCR is a reactor that is connected in series with back to back thyristor switches. The inductive value of the TCR can be continuously changed by changing the firing angle applied to its thyristors. Figure 2.2 illustrates the schematic diagram of a TCR and its instantaneous voltage and current waveforms. It is shown in Figure 2.2 that the voltage across a TCR is sinusoidal while the current through it contains odd harmonics.

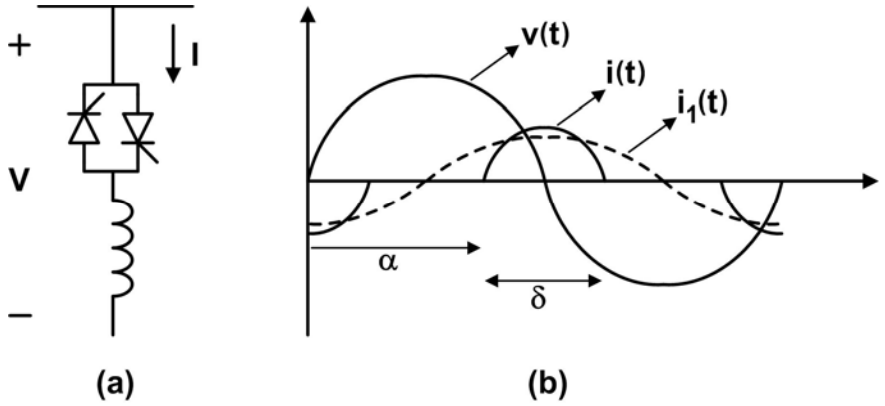

Figure 2.2. TCR, (a) schematic diagram, (b) voltage and current waveforms.

By applying Fourier analysis, the rms value of the fundamental component i_1 of the TCR current can be expressed as (Hingorani and Gyugyi, 1999):

$$I_1(\delta) = \frac{(\delta - Sin\delta) \times V}{\pi . X_L}, \tag{2-1}$$

where δ is the TCR conductance angle in radians and X_L is the fixed value of the TCR series reactor. The fundamental admittance B_1 of the TCR can be found by dividing the fundamental current component by the voltage across it.

$$B_1(\alpha) = \frac{2 \times (\pi - \alpha) + Sin(2\alpha)}{\pi . X_L}, \tag{2-2}$$

where α is the firing angle of the TCR switches. Hence, a TCR behaves like a controllable susceptance, which is a function of the switches' firing angle α. The nonlinear susceptance of a TCR and its *V-I* characteristics are shown in Figure 2.3. The value of the susceptance $B_1(\alpha)$ is normalized based on the value of $1/X_L$. By applying different firing angles to a TCR, a constant voltage region (line AB in Figure 2.3.b) can be achieved. Clearly, when α reaches 90^o the TCR is operating in full conductance.

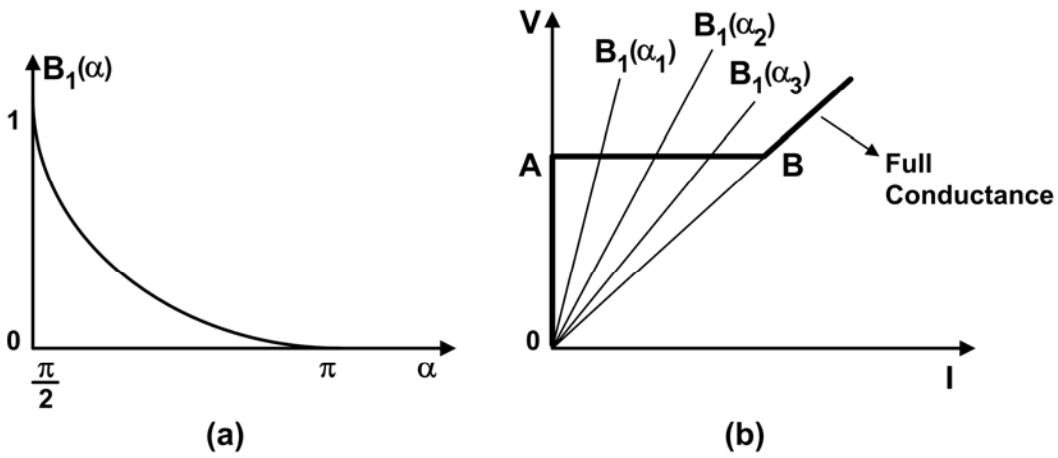

Figure 2.3. TCR, (a) nonlinear susceptance, (b) *V-I* characteristics.

2.2.1.2. Thyristor Switched Capacitor (TSC)

A schematic diagram of a TSC is shown in Figure 2.1. It consists of a fixed capacitor that is connected in series with two switches connected back to back. In order to avoid sudden current impulses, the TSC should be fired when the initial voltage across the capacitor is equal to the system voltage. Therefore, in any cycle there are only two suitable firing angles. When the thyristors are fired, they should be kept in the on position by applying continual gate pulses.

The switching speed for a TSC is much faster than a mechanical switch. However, even in this case a small time delay is inevitable, resulting in a small mismatch between the capacitor voltage and the line voltage at the PCC, which in turn causes a current impulse. This can be avoided by placing a small inductance in series with the capacitor.

2.2.1.3. SVC Configurations

The two more common configurations of a SVC are combinations of TCR-FC and TCR-TSC. Figure 2.4 illustrates the typical *V-I* characteristics of a SVC with the TCR-FC configuration.

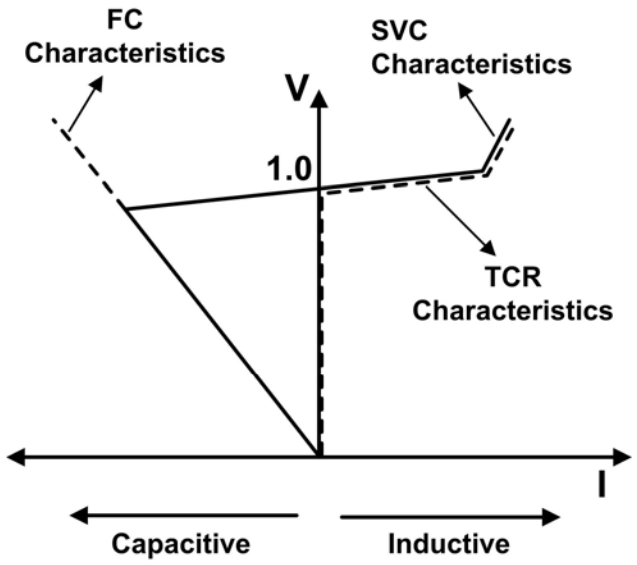

Figure 2.4. *V-I* characteristics of a SVC with TCR-FC configuration.

The operating region of the SVC can be both capacitive and inductive. The maximum reactive power in the capacitive region is equal to the MVar of the capacitor, while the maximum reactive power in the inductive region is equal to the sum of the MVars of the capacitor and the TCR. Therefore, the MVar rating of the TCR is higher than that of the SVC, since the TCR is always biased with a fixed capacitor. In a more accurate analysis the impedance of the filters and also the step down transformer should be taken into account as well. Figure 2.4 shows that the SVC has a droop characteristic that defines the per unit voltage error for the values of the SVC reactive current within the operating range of the device. This steady state voltage error can be avoided at the expense of

installing a very large capacitor, which can help maintain a constant voltage for different operating conditions of the power system. However, this requires a very large capital investment.

The performance of a SVC with the TCR-TSC configuration is slightly different. When operating in the inductive region, the TSCs are turned off. As the SVC starts operating in the capacitive region, the TSCs are turned on one by one if necessary.

2.2.2. Static Compensator (STATCOM)

Essentially a STATCOM is an ac/dc inverter connected to the power network in parallel. Two general structures exist for a STATCOM based on the type of inverter used: a voltage source inverter (VSI) or a current source inverter (CSI) STATCOM. The former is selected in this work due to its simpler structure and control. Figure 2.5 shows the schematic diagram of a VSI STATCOM connected to a power system. It is an inverter connected to a power system through a step-up transformer and a connection filter.

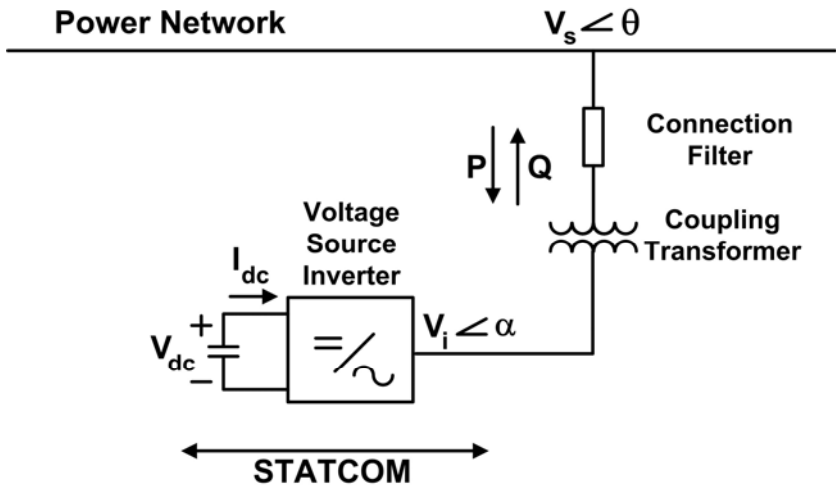

Figure 2.5. Schematic diagram of a VSI STATCOM.

By controlling the magnitude and the angle of its output voltage, i.e. $V_i \angle \alpha$, the STATCOM is able to control its active and reactive exchange with the power system and therefore control the voltage at the PCC.

$$P = \frac{V_i . V_S}{X_{Tr}} Sin(\alpha - \theta), \tag{2-3}$$

$$Q = \frac{V_i . (V_i - V_S . Cos(\alpha - \theta))}{X_{Tr}}. \tag{2-4}$$

where X_{Tr} is the leakage reactance associated with the step-up coupling transformer and the connection filter; it is a relatively small value around 0.1-0.15 p.u. (Hingorani and Gyugyi, 1999). The direction of the active power flow shown in Figure 2.5 is based on the assumption that $\alpha > \theta$.

Ideally, there are no losses in the STATCOM circuit; therefore, there is no active power exchange between it and the power system, and the capacitor voltage remains constant. However, in practice, the STATCOM has some switching losses associated with its semiconductor switches, and these are supplied by the dc link capacitor. This will gradually discharge the capacitor voltage. Therefore, the angle of the inverter output voltage α should be slightly lagging behind the angle of the voltage at the point of common coupling (PCC), i.e., the angle θ. The active power absorbed from the network as a result of this phase shift compensates for the STATCOM switching losses and maintains the dc link voltage at a constant value.

The direction of the reactive power flow can now be determined by the magnitude of the inverter output voltage. For values of V_i larger than V_S, the STATCOM is in the capacitive mode and injects reactive power into the network, while for the V_i values smaller than V_S, it is in the inductive mode and absorbs reactive power from the network.

In a typical STATCOM with a capacitor as the dc link, the values of the active and reactive power depend on one another. However, a STATCOM connected to a battery energy storage system (STATCOM-BESS) is capable of controlling the values of the active and reactive power independently (Yang *et al.*, 2001).

Typical *V-I* characteristics of a STATCOM are depicted in Figure 2.6 (Mathur and Varma, 2002), which show that the STATCOM can provide full capacitive/inductive current irrespective of the ac system voltage. This capability is specifically important during severe power system faults when parts of the power system are impaired and are unable to restore the required voltage. A STATCOM connected to such a system can improve the transient stability of the network, since it can behave like a current source. The current injected by the STATCOM into the network in both capacitive and inductive regions is limited due to the current ratings of the inverter switches. The points denoted as $I_{C\max}$ and $I_{L\max}$ in Figure 2.6 represent the STATCOM's capacitive and inductive limits during steady state conditions.

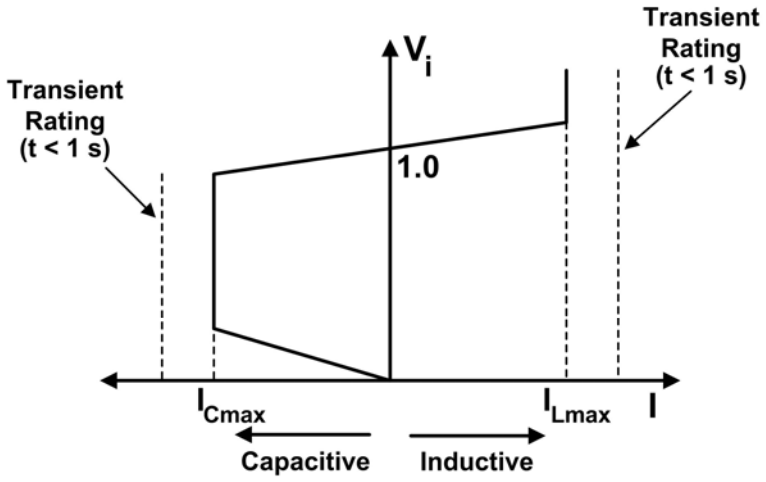

Figure 2.6. *V-I* characteristics of a STATCOM.

Figure 2.6 shows that the STATCOM controller is normally set to have a regulation slope. Instead of performing as a perfect terminal voltage regulator, the device is now allowed to vary the terminal voltage in proportion with the compensating current. This regulation droop extends the linear operating range of the compensator and tends to enforce the automatic load sharing between the STATCOMs and other voltage regulating devices in the network. Moreover, without this regulation droop, the system is prone to having a poorly defined operating point, as well as a tendency of oscillation (Hingorani and Gyugyi, 1999).

It can also be observed in Figure 2.6 that the STATCOM has an increased transient current limit in both capacitive and inductive regions compared to the steady state current limit. The maximum transient overcurrent in the capacitive region is determined by the maximum current turn-off capability of the converter switches. In the inductive region, the converter switches are naturally commutated; therefore, the transient current rating of the STATCOM is limited by the maximum allowable junction temperature of the converter switches (Mathur and Varma, 2002).

2.2.3. Advantages of the STATCOM

The latest developments in the new family of the high power fast switching insulated gate bipolar transistors (IGBT) supports the application of the STATCOM with fast controls at the transmission level (Yang *et al*., 2001). Furthermore, the STATCOM has the following advantages over the mechanically switched reactive compensators and the commonly used static Var compensator (SVC) (Mathur and Karma, 2002; Hingorani and Gyugyi, 1999):

- It occupies a small area compared to the bulky passive capacitive/inductive elements, since it replaces the actual reactive component with power electronics switches,
- It has a significantly faster response time compared to that of the SVC (200-350 μs compared to 2.5-5.0 ms),
- As opposed to an SVC, a STATCOM fed from an energy storage device, such as a battery, has the capability of exchanging active power with the network.

2.3. STATCOM Steady State Model

Figure 2.7 shows the single line diagram of the steady state equivalent circuit of a STATCOM connected to a power system (Schauder and Mehta, 1993). Values of R_{dc} and R_{Tr} represent the capacitor/switching losses and the coupling transformer losses respectively.

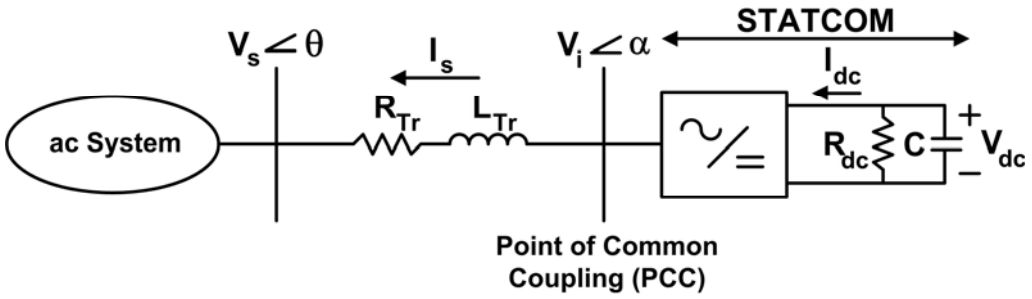

Figure 2.7. STATCOM single line equivalent circuit at steady state.

The first-order differential equations for the ac-side circuit of the STATCOM can be written as Equation (2-5). Similar equations apply to phases *b* and *c*.

$$\frac{di_{Sa}}{dt} = \frac{1}{L_{Tr}}(V_{ia} - V_{Sa} - R_{Tr}.i_{Sa}), \tag{2-5}$$

and the STATCOM dc-side equation can be written as:

$$\frac{dV_{dc}}{dt} = \frac{-1}{C}(I_{dc} + \frac{V_{dc}}{R_{dc}}). \tag{2-6}$$

In general the net input power provided by the dc-side should be equal to the net output active power provided by the ac-side, which is a small non-zero amount:

$$P_{in}^{dc} = P_{out}^{ac}. \tag{2-7}$$

Moreover, the reactive power at zero frequency (dc) is by definition zero; therefore, the capacitor has no role in the reactive power generation. This means that the converter simply interconnects the three ac terminals in such a way that the reactive output currents can flow between them, i.e., it provides a circulating current path (Hingorani and Gyugyi, 1999). The role of the dc link capacitor is to maintain a constant dc voltage for the operation of the voltage source inverter.

In practice, the output voltage waveform from the STATCOM is not a pure sinusoid, even if the current is close to a pure sine wave. This in turn causes the output active power to fluctuate around a constant value. The equality of the instantaneous input and output active powers requires that a fluctuating (ripple) current is drawn from the dc-side of the inverter. The size of the dc-side capacitor is chosen in such a way that the dc voltage across its terminals remains fairly constant irrespective of the dc current ripples (Mathur and Karma, 2002).

A more detailed explanation of the steady state performance of the STATCOM can be found in (Schauder and Mehta, 1993).

2.4. Applications of STATCOM in a Power System

The STATCOM is connected in parallel to a power network and is predominantly used for controlling the voltage at the point of connection to the power system. However,

various objective functions can be met by sending the appropriate control commands and designing a suitable control structure for the STATCOM, namely:

- Dynamic voltage and reactive power control at the transmission and distribution levels (Dong *et al.*, 2004; Mathur and Karma, 2002; Yang *et al.*, 2000; Hingorani and Gyugyi, 1999),
- Low frequency power oscillation damping at the transmission level (Mathur and Karma, 2002; Hingorani and Gyugyi, 1999),
- Torsional oscillation damping and subsynchronous resonance mitigation (Mathur and Varma, 2002; Patil *et al.*, 1998),
- Voltage flicker control (Sun *et al.*, 2002; Chen and Joós, 2001),
- Active power control, using energy storage systems (Dong *et al.*, 2004; Yang *et al.*, 2001),
- Voltage unbalance compensation (Valderrama *et al.*, 2001; Hingorani and Gyugyi, 1999).

Some of the most common applications of a STATCOM are briefly reviewed in this section.

2.4.1. Voltage/Reactive Power Control

The main objective of a STATCOM is to regulate the voltage at the point of connection to the network. As a rule of thumb, the best location for connecting a shunt FACST device in a power system is at the bus with the lowest voltage profile. In some specific cases where there is a high voltage drop across a transmission line, the optimum location for installing the STATCOM might be at the middle of the transmission line.

Regulating the voltage indirectly regulates the reactive power transmitted through the transmission line (Equation (2-4)) and affects the voltages of the neighboring buses and therefore, the power flow in the system. It is also possible to directly control the reactive power exchange between the STATCOM and the network, thereby indirectly controlling the voltage at the PCC.

2.4.2. Transient Stability

In general, regulating the voltage at particular points in the network, such as in the middle of the transmission lines and at specific load buses, increases the damping of the network. Moreover, with an appropriate and fast control scheme, a STATCOM is capable of providing voltage support during severe faults and disturbances. This further increases the transmission capability of the power system after the fault.

In particular, the efficiency of a STATCOM connected to the midline during transient disturbances can be explained using the equal area criterion (Glover and Sarma, 1986). Figure 2.8 shows a doubly fed power system with a STATCOM connected to the middle of the transmission line.

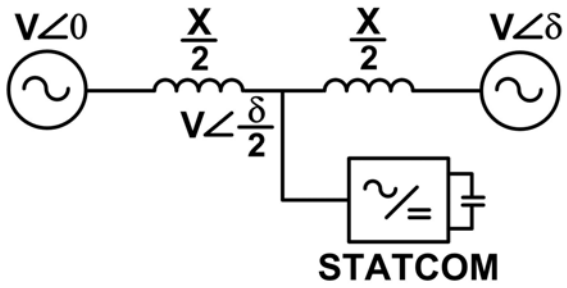

Figure 2.8. A doubly fed power system with a STATCOM.

The voltage sources at both ends of the line are assumed to have the same magnitude V, with an angle difference of δ. The transmission line is assumed to be lossless with a

total series reactance of X. For such a system with no reactive shunt compensation, the transmitted power through the transmission line can be expressed as Equation (2-8).

$$P = \frac{V^2}{X}.Sin(\delta) = P_{\max}.Sin(\delta)\,. \tag{2-8}$$

However, a STATCOM installed at the midline point can keep the voltage magnitude at the point of connection constant at V, with an angle of $\delta/2$. The total transmittable active power through the network can now be written as in Equation (2-9).

$$P = \frac{2\times V^2}{X}.Sin(\frac{\delta}{2}) = 2P_{\max}.Sin(\frac{\delta}{2})\,. \tag{2-9}$$

The $(P-\delta)$ diagram of the uncompensated and compensated power systems in Figure 2.9 clearly shows that the STATCOM is able to drastically increase the maximum transmittable active power. In the event of a severe fault, this helps decelerate the rotor and improve the transient stability during the first swing of the rotor angular velocity oscillation.

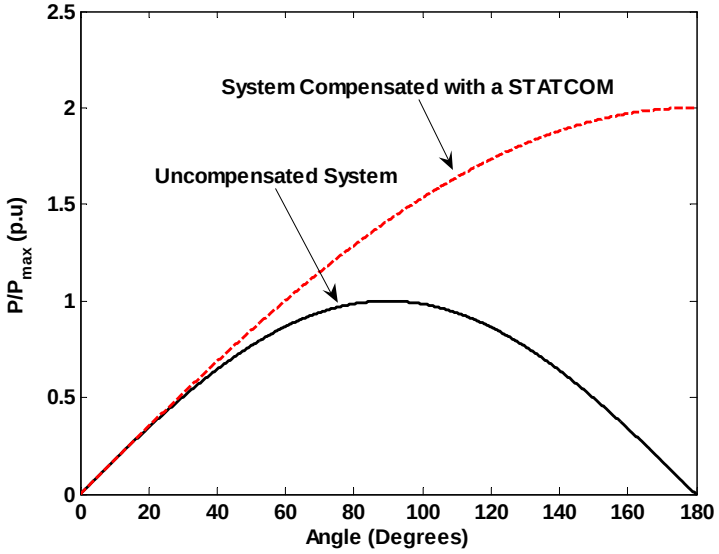

Figure 2.9. $P-\delta$ diagram of an uncompensated and a STATCOM compensated power system.

2.4.3. Dynamic Stability

For an underdamped power system any disturbance can cause low frequency oscillations in the rotor speed, electrical angle and the transmission line transmitted power. In general, the *swing equation* of a given machine can be written as:

$$P_m - P_e = 2H \frac{d^2\delta}{dt^2}, \tag{2-10}$$

where P_m and P_e are the mechanical power supplied by the prime mover and the electrical output power of the generator respectively, δ is the rotor angular position with respect to a synchronously rotating reference, and H is the synchronous generator inertia constant (Glover and Sarma, 1986).

In order to improve the damping of the low frequency oscillations, a compensator should counteract the accelerating and decelerating swings of the disturbed machine. That is, when the generator accelerates and δ increases, the compensator should increase the transmitted electric power to compensate for the excess mechanical input power. Conversely, when the generator decelerates and δ decreases, the compensator should reduce the transmitted power (Hingorani and Gyugyi, 1999).

However, Wang showed that the STATCOM dc voltage regulator contributes negative damping to the power system, while the ac voltage control has little influence on the system damping (Wang, 1999). Therefore, an auxiliary stabilizing signal should be added to the STATCOM ac voltage reference to combat the negative damping effect. The details are explained in the next section.

2.5. STATCOM Control Structure

Any STATCOM controller comprises of two main parts: an *internal controller* which, based on its reference command, sends out the appropriate firing signals to the inverter switches, and an *external controller* that processes the system information and generates the appropriate command signal for the internal controller.

2.5.1. STATCOM Internal Controller

Neglecting the voltage harmonics generated by the inverter, the output voltage of the STATCOM can be written as:

$$V_i = k.V_{dc}.Cos(\alpha - \theta), \tag{2-11}$$

where k is a constant and $(\alpha - \theta)$ is the difference between the angle of the inverter output voltage and the angle of the line voltage at the PCC.

Two main approaches exist in the literature for generating the inverter output voltage V_i (Hingorani and Gyugyi, 1999; Schauder and Mehta, 1993):

The *indirect method*, in which a square-wave pulse switching is applied, in which k is kept constant and the inverter output voltage is regulated by controlling the angle α. In this case the rms value of the fundamental component of the STATCOM output voltage can be written as (Mohan *et al*., 1995):

$$V_i = \frac{\sqrt{2}}{\pi}.V_{dc}.Cos(\alpha - \theta), \tag{2-12}$$

and the *direct method*, in which pulse width modulation (PWM) techniques are normally applied to control both the factor k and the angle α. The rms value of the fundamental component of the STATCOM output voltage can therefore be written as (Mohan *et al*., 1995):

$$V_i = \frac{m_a}{2\sqrt{2}}.V_{dc}.Cos(\alpha - \theta), \quad (2\text{-}13)$$

where m_a is the modulation index of the inverter.

In the indirect method, the angle α is controlled in order to regulate the dc link voltage V_{dc}, and indirectly the output voltage V_i. However, in the direct method the angle α controls the dc link voltage, while the output voltage is controlled by regulating the inverter modulation index m_a. Although both the angle and modulation index can be controlled in the direct method, these cannot be independently adjusted in steady state, since a traditional STATCOM has no significant active power capability (Yang *et al*., 2001).

2.5.2. STATCOM External Controller

The main objective of the STATCOM is to control the voltage at the point of connection to the network. However, with the introduction of auxiliary signals superimposed on the main reference signal, the STATCOM is able to provide peripheral control performance, such as transient stability (first swing), dynamic stability (damping), subsynchronous resonance damping, var reserve control and suchlike (Hingorani and Gyugyi, 1999). Such an approach can change the STATCOM performance from a local controller to a semi-global controller.

In the indirect method, the main reference signal can be a voltage command or reactive power command. However, in the direct method, a second control command for the dc link voltage is added to the STATCOM control structure. The two reference signals can go through either coupled or decoupled control loops.

Figure 2.10 shows the schematic diagram of the STATCOM controller. A direct method is assumed in which two reference signals are introduced for the STATCOM: line voltage reference V_{ref} and the dc link voltage V_{dc-ref}. A decoupled approach is chosen for the external control loops in which each of the control loops has their own PI controllers. In general, the dc link voltage and the inverter output voltage are not independent variables, so more sophisticated coupled control schemes exist in the literature that have taken this into account (Schauder and Mehta, 1993). The signals $\Delta\omega_m$, Δf and P in Figure 2.10 represent the rotor speed deviations and frequency deviations of the generator, and its output active power.

The switching module shown in Figure 2.10 computes the modulation index m_a and the inverter phase shift α based on the outputs of the PI controllers. The values of m_a and α are then passed into the STATCOM internal controller in order to derive the appropriate firing signals for the inverter switches.

Auxiliary inputs can be provided for the STATCOM in order to enable the compensator to perform peripheral control actions. Changes in the power system frequency, transmitted active power or the transmission line current are considered as appropriate signals for providing transient/dynamic stability auxiliary signals (Hingorani and Gyugyi, 1999; Mathur and Karma, 2002). Moreover, the deviations of the line frequency or the generator speed can be used in order to provide damping during the subsynchronous resonance (Mathur and Karma, 2002). Each of these auxiliary signals can be superimposed on the original ac voltage reference, or can be directly applied to the output of the ac voltage PI controller. The latter avoids time constants associated with

voltage regulator loop and improves the response time. Such an approach is particularly efficient for bang-bang control scheme (Hingorani and Gyugyi, 1999).

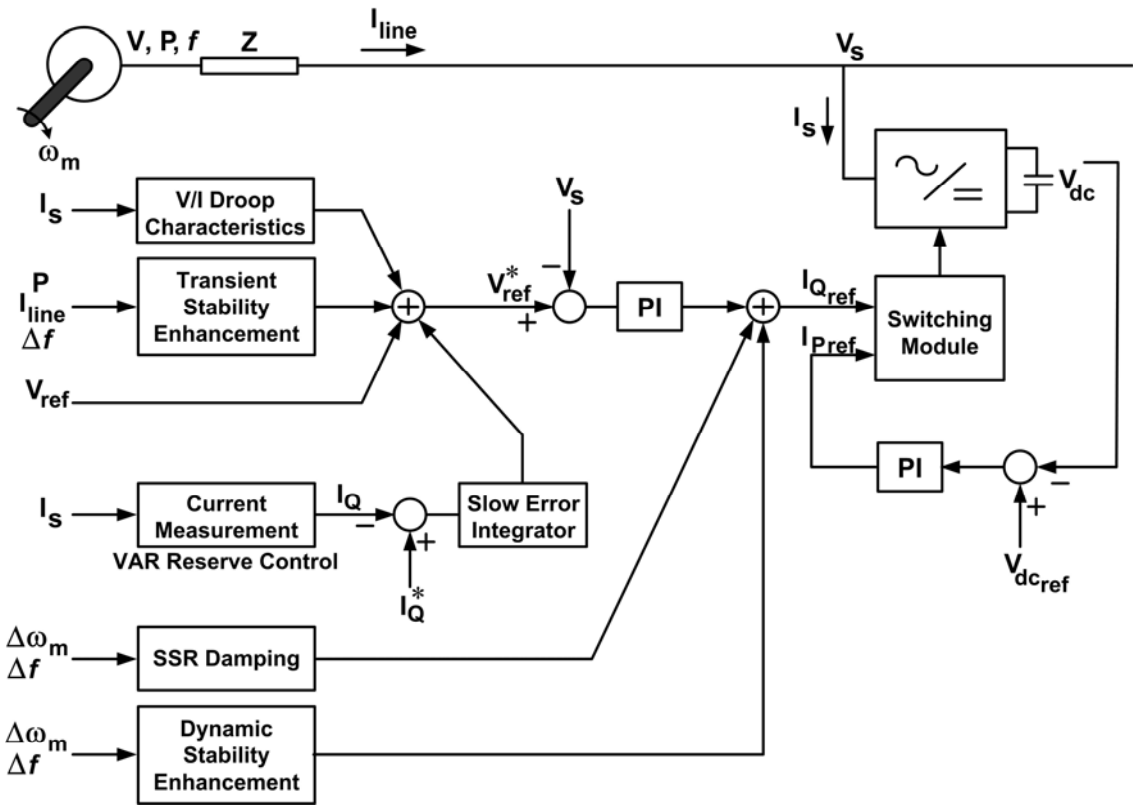

Figure 2.10. Schematic diagram of the STATCOM controller.

Some STATCOMs are equipped with a so-called Var reserve control loop. Sometimes it is necessary to ensure that the STATCOM has sufficient Var capacity to respond to unpredictable dynamic disturbances. This is often accomplished with an automatic control that maintains the predetermined Var reserve by adjusting the operating point of the STATCOM. It effectively changes the voltage reference in order to bring back the Var output slowly to the set reference value, and thereby activating slow Var sources such as mechanically switched capacitors and other means of var generation like the generator excitation to respond to the steady state var load (Hingorani and Gyugyi,

1999). The corresponding integrator should have a slow time constant so that it does not affect the performance of the system during the faults and disturbances, but has an impact on the steady state performance of the STATCOM.

2.6. Benchmark STATCOM Conventional Controller

The main objective of the STATCOM in this study is to control the voltage at the point of connection to the power system. Figure 2.11 shows the schematic diagram of the STATCOM conventional controller used hereafter as a basis for comparison of the performances of the proposed fuzzy and neural network based controllers.

The proposed controller utilizes a decoupled direct method for the dc link voltage and the line voltage control loops. The line voltage at the PCC is compared with the reference signal, the corresponding error is passed through a PI controller whose output determines the inverter modulation index m_a. The dc link voltage error is also passed through a second controller and generates the inverter phase shift in the output voltage α.

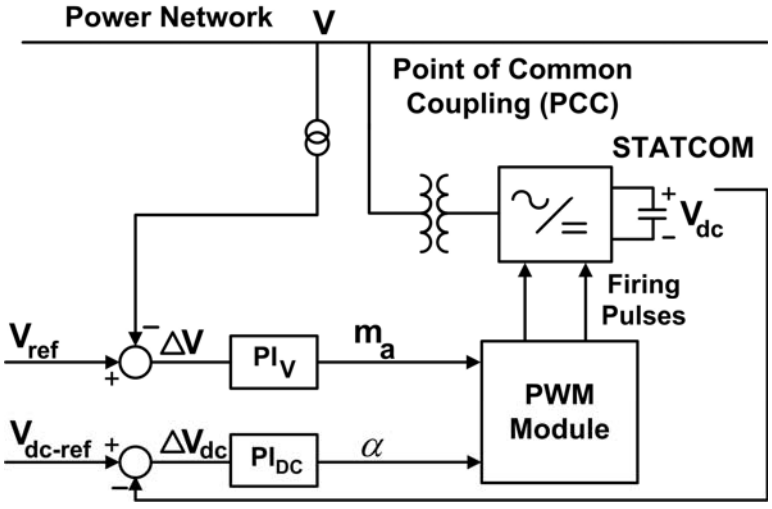

Figure 2.11. STATCOM conventional controller.

This scheme is compared with the coupled direct scheme suggested by Dong *et al.* (2004) and proves to be more efficient in damping the power system oscillations, as well as following a step change in the reference values of the STATCOM.

The PI controllers in Figure 2.11 are fine tuned at one operating point by trial and error to ensure that they have satisfactory performance in that point (details appear in Appendix F). When tuning a PI controller it is important to know that there is always a tradeoff between the satisfactory performance during the transients and during the steady state. A PI controller can be fine tuned to show excellent performance during small scale disturbances that do not cause drastic changes in the system operating conditions, such as a small scale step change in the reference signals. While such a controller might show near optimal performance (fast response, no overshoot) for a small scale disturbance, it will fail to perform equally well during a more severe fault, such as a three phase short circuit, since the fault drastically changes the operating conditions of the nonlinear power network.

2.7. Summary

A STATCOM is a power electronic converter that is connected to the power system in parallel and can mainly control the line voltage at the point of connection to the power network. By providing auxiliary signals, a STATCOM is also able to provide dynamic and transient damping for the power system. In this chapter the basics of the STATCOM, its modeling, applications and typical control structure were explained in detail. Moreover, a decoupled control structure was proposed for the STATCOM using conventional PI controllers, which hereafter is used as the benchmark controller to be compared with the intelligent controllers designed for the STATCOM (Chapters 5 and 6).

CHAPTER 3

LITERATURE SURVEY

3.1. Introduction

In this chapter a summary of the previous work on the local and global control of a power system is presented. The focus of the next sections is on the research conducted on the following:

- Internal (local) control schemes for a STATCOM in a power system,
- External control schemes for a STATCOM in a power system,
- Wide area (global) control schemes for a multimachine power system.

In the next sections, the term internal controller, also referred to as local controller, indicates that the corresponding controller tries to regulate a local quantity, such as the voltage or power, based on the local measurements available. The term external controller refers to the fact that the controller is designed to achieve a peripheral control objective by observing and regulating a second variable in the power system rather than the main control variable. The actions taken by the external controller are at times in contrast to the internal controller objective. However, an external controller is normally active only during transients or other short term disturbances, while during steady state condition, the internal controller is in charge of controlling the component.

Both internal and external controllers control a single component. A wide area controller (WAC), on the other hand, emphasizes on the fact that the controller is sending the control commands to various components in the power system. Similar to the external controller, the WAC uses remote signals in addition to the local signals in order to

achieve its control objective. However, a WAC usually covers a wider geographical area in a power network than an external controller.

In a hierarchical control regime, an internal controller is in the primary level of control, whereas the external controller and WAC represent the second level of control by closing the loop of the internal controllers.

3.2. STATCOM Internal Control

3.2.1. Linear Control Techniques

Linear techniques have been predominantly used for controlling a STATCOM. In this approach the system equations are linearized around an operating point. Based on this linearized model, the conventional proportional-integral (PI) controllers are fine tuned to effectively respond to the small scale and large scale disturbances in the power system, where the STATCOM is connected to.

Schauder and Mehta (1993) developed two separate linear control techniques for a voltage source inverter (VSI) STATCOM for regulating the output voltage magnitude and/or the phase angle. Wang (1999) designed a linear controller for a STATCOM in a single machine infinite bus using the linearized Philips-Heffron model of the power system. Similarly, Shen *et al.* (2000) proposed a per-unit mathematical model for a STATCOM in a power system by linearizing the system equations. Also, Rao *et al.* (2000) designed a linear PI controller for a STATCOM in a power system and they further improved its performance by a pole placement method.

Decoupled or semi-decoupled linear control structures have also been proposed in the literature for regulating the active and reactive power exchange between the STATCOM and the power system. Some researchers have proposed decoupled linear controllers to

control the STATCOM active and reactive power injections independently (García-González and García-Cerrada, 2000). Others have designed semi-decoupled linear controllers for this purpose (Yang *et al.*, 1998; Dong *et al.*, 2004). In both cases, the STATCOM was connected to a multimachine power system and the linearized state space equations were used for implementing the controller structure. Dong *et al.* (2004) also validated the controller performance by hardware implementation.

In all those control schemes, the voltages are indirectly controlled by regulating the active and reactive power exchanged between the STATCOM and the rest of the power system. Dong *et al.* (2002) suggested a linear control structure, consisting of two loosely coupled PI controllers, for directly regulating the line voltage at the PCC and the dc link voltage inside the STATCOM.

Linear controllers have also been designed for the STATCOM that address control objectives rather than regulating the line voltage and/or the active and reactive power flows. For instance PI controllers were used to design internal controllers for distribution STATCOMs that enabled them to mitigate voltage flicker (Sun *et al.*, 2002; Chen and Joós, 2001).

The main disadvantage of the above linear models/controllers is that they are determined under certain restrictive assumptions, such as linearizing a nonlinear system and/or approximating a higher order system by a low order model. While these models are appropriate for certain small signal applications in the vicinity of a specific steady state operating point, they cannot reflect the true nature of the power network and the STATCOM when the system is exposed to large scale faults or dynamic disturbances that change the configuration of the plant to be controlled.

Moreover, in most of the industrial applications, the parameters of the PI controllers are fine tuned by trial and error. Various methods exist in the literature that focus on effective ways to design PI controllers. However, the majority of these techniques are somehow dependent on a mathematical model of the plant to be controlled (Krajewski *et al.*, 2004; Clarke, 2003; Kristiansson and Lennartson, 2002; Suchomski, 2001; Poulin and Pomerleau, 1996), which is often only approximate in any case. Even under such conditions the solution will not necessarily be trivial, and sometimes uncertainties associated with real life problems further exacerbate the reliability of such approaches.

3.2.2. Nonlinear Control Techniques

The disadvantages of the linear controllers/models, such as being dependent on the operating condition, can be overcome by using appropriate nonlinear control techniques. The well-established theory of nonlinear controllers range from fixed structure controllers to more complicated adaptive, robust and optimal designs (Khalil, 2002; Harris and Billings, 1985; Bertsekas, 2000).

Many researchers have proposed nonlinear controllers for a STATCOM. Petitclair *et al.* (1996) designed a nonlinear control based on the exact linearization via feedback. Another nonlinear controller was designed for a STATCOM based on the differential algebra theory (Yao *et al.*, 1998). The proposed controller was successfully tested by evaluating its performance when applying step changes to its reference signals. The generalized averaged method was used to obtain a continuous nonlinear model of the power system with the STATCOM. Lu *et al.* (2001) designed a robust nonlinear controller based on the approach of nonlinear disturbance attenuation in the sense of L_2-gain. The controller could effectively control the STATCOM in the presence of the

disturbances applied to the generator torque and the STATCOM output voltage. Valderrama *et al.* (2001) implemented a nonlinear adaptive controller, following the passivity based design procedure that was able to deal with design parameter uncertainties. The controller showed successful performance during unbalanced conditions. Shen and Lehn (2002) designed a STATCOM nonlinear controller that had two loops: a slower outer loop for controlling the dc link current and a faster inner loop, using integral control, for controlling the ac current. Liu *et al.* (2003) implemented a robust nonlinear controller based on the H_∞ design approach for a STATCOM in a single machine infinite bus system.

While most of the above techniques provide good control capability over a wide range of operating conditions, they have some disadvantages associated with them, namely all these designs have a more sophisticated structure and are more difficult to implement compared to the linear controllers. More importantly, they also need a mathematical model of the power system to be controlled. However, developing an analytical model for a power system becomes less practical for a large real world power network with numerous interconnections. This is a significant drawback for the nonlinear design techniques in multimachine power systems.

3.2.3. Intelligent Control Techniques

Several researchers applied intelligent control techniques, such as fuzzy logic, to solve the problems associated with the linear and nonlinear STATCOM controllers. Fuzzy logic can compensate for the potential problems associated with analytical linear/nonlinear controllers. It was proposed by Lotfi Zadeh in the early 60's as a way of characterizing non-probabilistic uncertainties, i.e., "*fuzzy sets*" (Zadeh, 1965). Fuzzy

logic is a technique that deals with imprecise, vague or “fuzzy” information. Fuzzy controllers consist of a set of linguistic control rules based on fuzzy implications and the rule of inference. By providing an algorithm, they convert the linguistic control strategy based on expert knowledge into an automatic control strategy (Lee, 1990). In contrast to the mathematical models or other expert systems, fuzzy systems allow the representation of imprecise human knowledge in a logical way, with approximate terms and values, rather than forcing the use of precise statements and exact values; thus making them more robust, more compact and simpler (Song and Johns, 1998).

Fuzzy logic based controllers are nonlinear controllers that do not necessarily need any prior information about the power system. They are generally independent of a mathematical model for the system to be controlled, which is a major advantage over the majority of other nonlinear controller designs. Moreover, they can provide efficient control over a wide range of system operating conditions, which distinguishes them from conventional linear controllers.

Fuzzy logic based control technique was used for designing a nonlinear controller for a STATCOM to improve the dynamic stability of the power system it is connected to (Li *et al.*, 1998). Also, Singh et al. (2000) implemented a fuzzy logic based controller in hardware for controlling a VSI STATCOM. Fuzzy systems were also used for designing a nonlinear controller to control the reactive component of the STATCOM current in a power system (Morris *et al.*, 2003). Chia *et al.* (2004) implemented a fuzzy logic based controller together with a feedback linearization controller for a STATCOM to improve the damping of the electromechanical oscillations in the power system.

In all the above fuzzy logic based controllers, the parameters of the controllers are fixed throughout their performances. This can sometimes be a disadvantage since the power system configuration and operating conditions change as the shunt loads or transmission lines are switched on/off. Although a fuzzy controller has an inherent robustness to parameter uncertainties, any change in the power network might at times degrade the controller's performance compared to the optimal status it is designed for. Also, the rule base of a fuzzy controller is derived heuristically based on the expert knowledge. An operating condition/topology of the power network that is not properly taken into account in the original rule base can reduce the effectiveness of the fuzzy controller.

Artificial neural networks have also been applied to improve the performance of STATCOM controllers. Mohaddes *et al.* (1999) applied neural networks for developing an algorithm to control the harmonic output of a STATCOM. The proposed switching scheme proved to be more efficient than the conventional PWM.

Table 3.1 summarizes the existing internal control techniques for a STATCOM, along with their corresponding advantages and disadvantages.

Table 3.1. Internal control schemes for a STATCOM.

Controller Type	PI	Nonlinear (Fixed Structure)	Nonlinear Adaptive	Fuzzy Logic Controller
Depends on the Operating Conditions	Strongly	No	No	No
Needs analytical model	Not essential	Yes	Yes	No
Is robust to changes in the power system configuration	No	Not necessarily	Yes	Partially

3.3. STATCOM External Control

Many researchers have investigated the possibility of enhancing the capabilities of the STATCOM by providing an additional control loop that sends out an auxiliary control signal to the STATCOM and tries to control it from a supervisory level.

Most of these proposed external controllers employ a linear control scheme, in the form of a simple proportional gain or a PI controller. Hingorani and Gyugyi (1999) proposed a PI based external control structure for the STATCOM that enabled it to increase the damping of the power system (Figure 2.6). Patil *et al.* (1998) designed a linear controller that provided auxiliary control signals to the line voltage control loop of a STATCOM connected to a single machine infinite bus. By applying this auxiliary signal to the line voltage reference of the STATCOM, it was able to damp out the torsional oscillations in the power system more effectively. Also, Mathur and Karma (2002) designed a linear external controller for a STATCOM to improve the damping of the low frequency power oscillations in the power system.

The inherent disadvantage of the linear control schemes, i.e., being dependent on the operating conditions of the power system, is not as critical for the STATCOM external control as it is in the case of its internal controller. This is due to the fact that the external controller does not normally have a role in the main control objective of the STATCOM. Rather, it is designed to improve the capabilities of the STATCOM mostly during the transients and/or dynamic disturbances. However, the efficiency of the linear external controller can still be affected by a change in the operating conditions of the power system or its topology. Moreover, as the complexities of the power system to which the

STATCOM is connected increases, linear control schemes become less efficient in achieving the desired supervisory level control objectives.

Some external controllers have been designed that incorporate nonlinear or intelligent control schemes. Farsangi *et al.* (2000) designed a H_∞ robust controller to improve the transient stability of a 4-bus 2-generator power system. Yixin *et al.* (1999) designed a fixed structure fuzzy logic based external controller for damping out the inter-area oscillations in a two-area multimachine power system. Qu and Chen (2002) also reported the successful application of the fuzzy logic based external controller, with a fixed structure, for a STATCOM in a similar application.

The same disadvantages mentioned for the nonlinear and fuzzy logic based internal controllers apply here as well. Any nonlinear external control scheme requires a mathematical model of the power system, which deriving it in most of the cases is a very tedious task.

On the other hand, fuzzy logic based external controllers with fixed parameters lose their efficiency as the dimensions of the power system increases, since there is no guarantee that a heuristic rule base, according to a human expert, can provide an optimal performance.

3.4. Wide Area (Global) Control in Power Systems

Several attempts have been reported in the literature for controlling a power system from a supervisory level.

Arafeh (1978) published one of the first papers on the concepts of the hierarchical control theory in power systems. He designed a hierarchical controller to enable the real-time control of a large scale power distribution system in terms of security (equipments

protection) and operation (continuity of service). In this approach, the overall control strategy for each substation was decomposed into independent control functions which were ranked (prioritized) to meet a hierarchy of control requirements specified by the system's physical operation. The control algorithm logic was implemented using heuristic programming (Arafeh 1978).

Rubaai and Villaseca (1989) proposed an optimal hierarchical controller for improving the transient stability of a multimachine power system. The control technique involves a number of independent local controllers that communicate with a central coordinating controller. Using a separable global cost function, the proposed controller provides global optimal transient performance (Rubaai and Villaseca, 1989). Okou *et al.* (2004) extended the previous work by designing a two level hierarchical controller to improve power system stability under severe contingencies. Their solution consisted of a local controller for each generator, which communicates with a central controller at the secondary level. Using remote signals from all the generators in the power network, the secondary level controller applies auxiliary control signals to the local controllers in such a way that it compensates the interactions among the generators, and therefore the inter-area oscillations (Okou *et al.*, 2004).

Posser *et al.* (1995) applied the methodology of discrete-event systems (DES) and supervisory control to the transmission line restoration problem, in order to increase the steady state security level of a power network during restoration. At each state of the power system, the proposed supervisory level controller defined an envelope of the allowable actions in terms of enabling/disabling the restoration of a transmission line.

Within that envelope, local controllers make specific decisions to satisfy local performance indices (Posser *et al.*, 1995).

However, much of the work in the past few years has focused on designing a global PSS. Aboul-Ela *et al.* (1996) designed a PSS that uses global signals in addition to the local control signals and can be used for damping inter-area oscillations as well as the ones caused by the local modes. It is essentially a two level PSS where the first level of control is derived from local signals and is effectively designed to deal with the local modes. The second level of control is supplied from a coordinator using selected global states to deal with the poorly damped inter-area modes. In their approach, they introduced a second control signal, i.e., the deviations of the power flow in a certain tie-line or the rotor speed difference between two generators, and showed that the PSS with global signals is effective in increasing the system damping for inter-area modes (Aboul-Ela *et al.*, 1996).

Chow *et al.* (2000) also incorporated the second input to the PSS from one of the neighboring generators, as an auxiliary control signal for the controller. The design was implemented in a small equivalent Brazilian system in which the unstable open loop system could not be stabilized by a single conventional PSS. Chow *et al.* showed that the system can be stabilized using a PSS with two input signals, i.e., a local and a remote signal (Chow *et al.*, 2000). Kamwa *et al.* (2001) applied a similar technique by incorporating additional remote signals to a selected number of PSSs in the Hydro-Quebec's transmission system. The power system stabilizers that had the highest effects on damping the inter-area oscillations were chosen based on a thorough singular value and eigenvalue analysis of the power system (Kamwa *et al.*, 2001). Also, Ni *et al.* (2002)

designed a supervisory level PSS by using wide area measurements. In their approach the local PSS on the generators are considered as local agents. A robust fuzzy logic based supervisory PSS receives remote signals, based upon which it sends out appropriate control signals to the local machines.

It has also been shown that with the introduction of auxiliary signals, FACTS devices can contribute to the dynamic and/or transient stability of the power system (Hingorani and Gyugyi, 1999). This issue has been investigated by some researchers in an attempt to design a supervisory level controller. Chow *et al.* (2000) showed that by introducing two machine speed measurements (instead of one local measurement) as control signals, the performance of a thyristor controlled series compensator (TCSC) damping controller is improved. Aboul-Ela *et al.* (1996) used a remote generator's speed deviations or tie-line power flow deviations as the second control signal in order to improve the effectiveness of a SVC to increase the dynamic damping of the power system. Wang (2001) utilized the multi-agent hierarchical control theory for secondary voltage control in a power system. In this scheme the supervisory level control closes the control loop of the reference settings of the controllers at the primary level, i.e., the local controllers. The objective here is to minimize the voltage violations in the power system. The "request and response" approach is selected for task sharing between the agents (AVRs, STATCOMs and SVCs in the power system). Each agent monitors the voltage and if it sees a violation, it tries to inject reactive power to counteract it. In case an agent fails to do its job, it selects agents in its agent system to send out a request to perform the task.

The complexity of a large power network often makes it very hard for an analytically based control technique to perform a supervisory level control of the system. Artificial

intelligent techniques on the other hand, have the capability of dealing with such a nonlinear, non-stationary system in the presence of noise and uncertainties. Neural networks, for example, have been used to design an optimal wide area stabilizer for providing extra damping for the generator rotor speeds (Venayagamoorthy and Ray, 2005). Using the adaptive critic designs (ACD) theory, the proposed controller was able to provide additional damping for the inter-area oscillation modes in a multimachine power system.

Also, Kim and Lee (2005) presented an artificial neural network based coordination control scheme for an under load tap changing (ULTC) transformer and a STATCOM. The objective of the coordination controller was to minimize both the amount of tap changes of the transformer and STATCOM output while maintaining an acceptable voltage magnitude at the substation bus.

3.5. Summary

A literature survey of the research carried out on designing controllers for a static compensator has been presented in this chapter. Various techniques, i.e., linear, nonlinear and intelligent control designs, applied for internal (local) and external (supervisory level) control of a STATCOM have been reported along with their corresponding advantages and disadvantages.

In addition, this chapter has reviewed the previous work done on wide area (global) control of power systems. Most of the research in this area has focused on employing remote signals for improving the damping capabilities of power system stabilizers or some FACTS devices, such as an SVC or a TCSC.

CHAPTER 4

A NEURAL NETWORK BASED IDENTIFIER FOR A STATCOM IN A MULTIMACHINE POWER SYSTEM

4.1. Introduction

A model of the plant/process to be controlled is the first requirement of most adaptive/optimal control techniques. For a simple system, a mathematical model can be expressed by a set of differential/difference equations. As the dimensions and the complexity of the plant increase, finding a set of linear/nonlinear equations which can truly describe the dynamics and behavior of the system becomes more difficult and less feasible. This is the case for almost all the applications in multimachine power systems.

The system considered in this chapter is a 9-bus 4-generator power system with a STATCOM (Figure 4.1). It is a 500 kV, 5000 MVA system and is simulated in the PSCAD/EMTDC® environment. Each of the generators is modeled together with an automatic voltage regulator (AVR), exciter, governor and turbine dynamics taken into account (Anderson and Fouad, 1994). The details of the power system are presented in Appendix B. A STATCOM is connected to bus 5 in order to provide extra voltage support for the load area. Hereafter, the *plant* consists of the generators, their controllers, transmission lines and loads, as well as the STATCOM, its PWM module and the dc link PI controller (PI_{DC}), while the *controller* corresponds to the line voltage PI controller (PI_V). The controller generates a control signal u (i.e., the inverter modulation index m_a) which is the plant input, and in turn controls the line voltage V at the point of common coupling (PCC), which is fed back as the plant output.

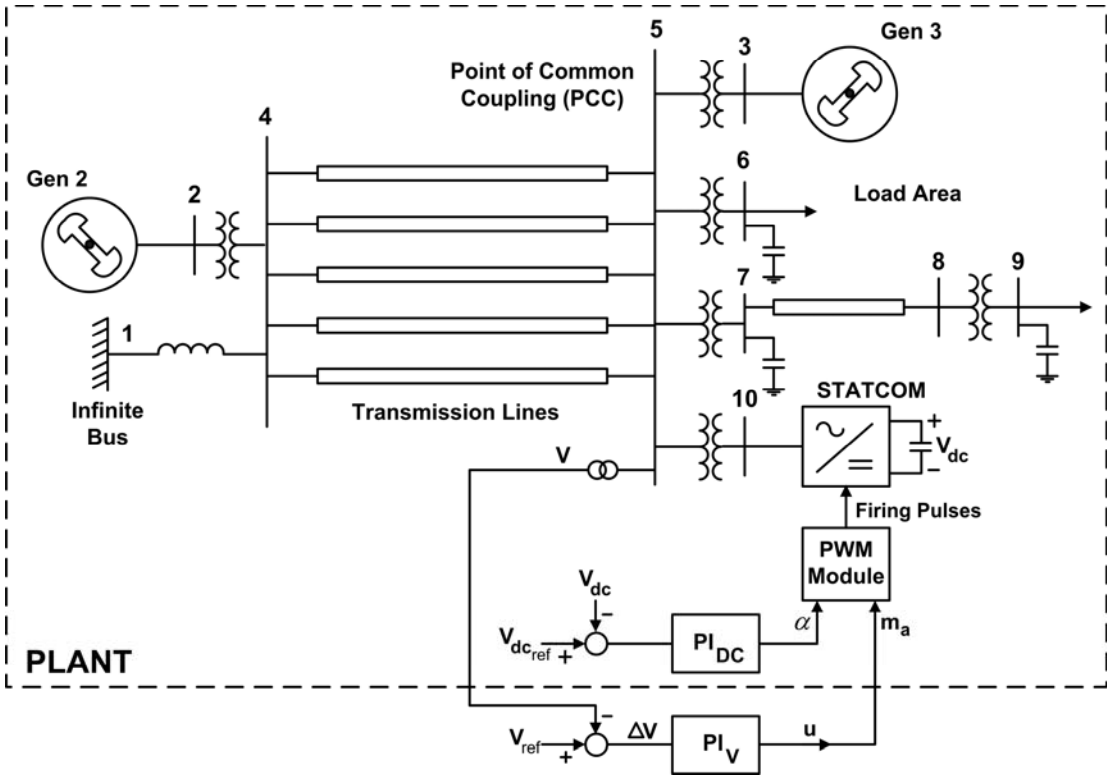

Figure 4.1. Multimachine power system with STATCOM.

Various neural network architectures exist in the literature that can be trained to identify a highly nonlinear plant such as a multimachine power system. Two most popular designs which are commonly used are multilayer perceptron (MLP) neural network and radial basis function (RBF) neural network structures (Narendra, 1992; Narendra and Parthasarathy, 1990) (see Appendix C). In this chapter a RBF based neuroidentifier is designed to model the plant in Figure 4.1. The objective of the proposed neuroidentifier is to learn the dynamics of the plant and track its output V during the small signal disturbances as well as large scale severe faults.

Generalization is the most important aspect of a neural network based identifier. It is a measure of how well the neural network can perform in an actual problem after sufficient training is done. Generalization is affected by the number of data samples (and

how well they represent the problem space), the complexity of the underlying problem and the network size (Hush and Horne, 1993). The first and third issues are discussed in detail in this chapter.

Another important aspect of training neuroidentifiers is related to the stability-plasticity dilemma, which investigates how a learning system can be designed to remain plastic or adaptable enough to learn new things whenever they might appear, and yet remain stable enough to preserve previously learned knowledge (Kartalopoulos, 1996). This issue is addressed in this chapter as well.

4.2. Why RBF Neural Networks?

In theory, both MLP and RBF neural networks are capable of forming an arbitrarily close approximation to any continuous nonlinear mapping; however, a RBF is more efficient in classification problems while a MLP is more effective in function approximation. This is due to the fact that RBF units (neurons in the hidden layer) generate localized response to the input stimulus. This means RBF units produce a significant nonzero response only when the input to the neural network falls within a small localized region of the input space (Hush and Horne, 1993). Therefore, each neuron in the hidden layer focuses on a specific region of input space. Moreover, a RBF network is efficient in dealing with both actual signals and deviations, while MLP fails to track actual signals (Haykin, 1998). This is an important advantage in system modeling applications, since a system model based on deviation signals is a small signal model, while a large signal model is the one that is really needed in power systems modeling applications.

In addition, it has been known that the inputs to the MLP neuroidentifier should always be normalized between -1.0 and +1.0 (Hush and Horne, 1993). Failing to do so will cause some of the inputs gaining more unrealistic importance over the other ones. For a small signal model that only learns the response of the plant to small disturbances, this does not create any problems. However, for a large signal model, where the response of the plant to both small scale and large scale disturbances are of interest, normalizing the whole neural network input space will bias the synaptic weight matrices towards the larger values and the network fails to learn the system dynamics during small scale disturbances. Clearly, this issue is not a major concern in a RBF network, due to the localized nature of the RBF units.

Several methods exist for deriving the centers of the RBF network (Uykan *et al.*, 2000). An offline clustering method is used in this study, according to which the centers of the RBF units are determined offline and are not updated during the network training (see Appendix C). If the offline input data samples are a true representative of the problem space, then a finite number of neurons in the hidden layer will suffice for modeling the system with reasonable accuracy. Moreover, this will prevent the neural network from forgetting the previously learned information (*stability* issue). The output weight matrix is the only network parameter which is updated online in order to force the system adapt to dynamics that were not considered during the training (*plasticity* issue). Clearly, the stability issue can not be easily achieved by a MLP, and such a network should undergo continuous training at all times.

4.3. Neuroidentifier Structure

The typical structure of the RBF neuroidentifier is shown in Figure 4.2. The network has seven neurons in the input layer, a single layer of neurons in the hidden layer and a single output neuron. The neuroidentifier receives the plant input u and plant output V at time steps $(t-1)$, $(t-2)$ and $(t-3)$; and in turn estimates the plant output at one step ahead, i.e., $\hat{V}(t)$. This is achieved by developing a nonlinear autoregressive mean average (NARMA) model of the plant to be controlled (Narendra and Parthasarathy, 1990). A Gaussian function is used as the activation function of the hidden layer neurons and the backpropagation algorithm is applied for updating the synaptic output weight matrix of the neuroidentifier (see Appendix C).

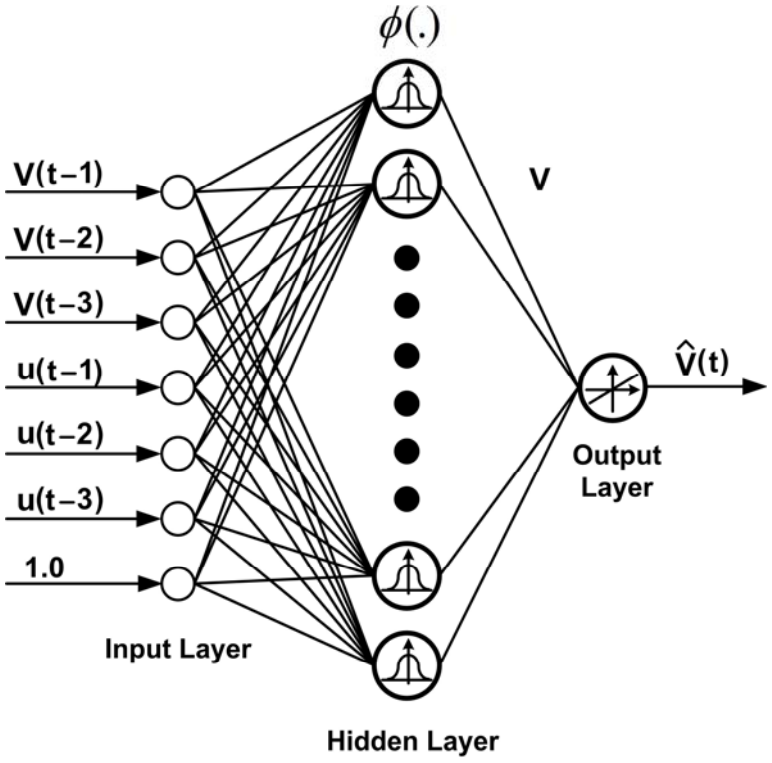

Figure 4.2. RBF neuroidentifier schematic diagram.

4.4. Size of the Neuroidentifier

Neurons in the hidden layer(s) of a neural network act as memory cells. While a network with a small number of neurons may not be able to model a given problem, increasing the number of neurons to a very large number on the other hand can make the neural network too capable (Baum and Haussler, 1989). This may have negative effects on the generalization properties of the neural network by over-fitting the data. Such a network will show good results on the training data, but might perform poorly on a separate set of data. Moreover, increasing the number of neurons will increase the computational effort of the network that can severely affect the simulation speed. Essentially there should be a tradeoff between the number of the neurons in the hidden layer and the computational and generalization efficiency of the neural network.

Several techniques exist in the literature in order to determine the optimum number of neurons in a neural network (Hush and Horne, 1993). One approach is to start the training with a very large network size. Such a network will have a large number of redundant information in its synaptic weights. Using *pruning* techniques, the weights and/or nodes which have little or no contribution to the solution are then discarded. Another possibility is *growing* a neural network by starting with a small number of neurons and gradually increasing them until little or no improvement is observed in the network performance.

In this chapter the latter technique is selected in order to determine the optimum number of neurons for the neuroidentifier. The neural network starts with two neurons and random input/output synaptic weight matrices. A training set with 20,000 data samples, referred to as the training epoch, is applied for training the network. After each training epoch the number of neurons in the hidden layer is increased by one and the

same training epoch is continued. The performance of the neural network is evaluated in terms of the mean squared error (MSE) defined by Equation (4-1):

$$MSE = \frac{1}{N}\sum_{i=1}^{N} | e(i) |^2 , \qquad (4\text{-}1)$$

where N is the number of data samples in each training epoch, i.e. 20,000, and $e(i)$ is the neuroidentifier tracking error for the i^{th} data sample. Figure 4.3 shows the final values of the MSE at the end of each training epoch are shown for the neuroidentifier. Figure 4.4 shows the normalized value of the MSE based on its initial value.

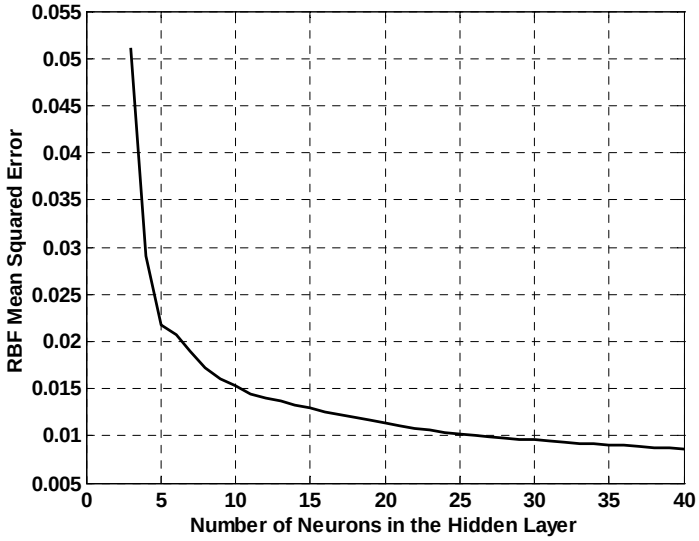

Figure 4.3. Mean squared error for different numbers of neurons.

The value of the MSE for each training epoch is compared with the value corresponding to its previous training epoch. An improvement of less than 0.5% in the MSE is adopted in this study as the stopping criteria for selecting the optimum number of hidden layer neurons. According to this, the optimum number of neurons for the RBF neuroidentifier is chosen to be 19.

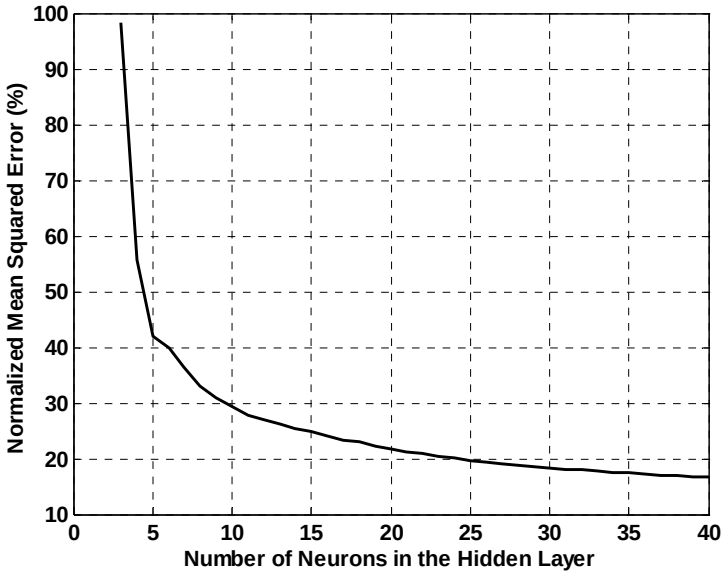

Figure 4.4. Normalized mean squared error for different numbers of neurons.

4.5. Computational Complexity Analysis

The size of the neural network directly determines the total number of the required mathematical calculations for each training sample. This number can serve as an additional measure for selecting the right size and architecture of the neural network. The simulation time required for executing the feedforward and backpropagation paths for each training sample is specifically important for online training applications.

Table 4.1 summarizes the number of mathematical operations for each data sample for the RBF neural network, assuming the neural network has *m* inputs, *n* neurons in the hidden layer (*n* RBF units) and *r* outputs. A non-zero momentum gain is assumed for updating the synaptic output weight matrix. Finding the centers offline (as in this study) requires additional calculations that are not included in Table 4.1.

Table 4.1. Mathematical calculations for a RBF neural network of size $(m \times n \times r)$.

ANN Architecture	RBF Neural Network Operations			
	$\times$	$+/-$	$\sqrt{\ }$	$e^{(.)}$
Decision Vector	$(n \times m) + 2n$	$n \times m$	0	n
Output Vector	$r \times n$	$r \times (n-1)$	0	0
Error Gradient at Output Neurons	0	r	0	0
ΔV	$3(r \times n)$	$r \times n$	0	0
Center Update	$(n \times m) + m$	$(n \times m) + 2m$	n	0
Center Width Calculation	$(n \times n \times m) + 1$	$(n \times n \times m) + n$	n	0
Weight Update	0	$r \times n$	0	0
Total Computations	$n.[m \times (n+2) + 4r + 2] + m + 1$	$n.[m \times (n+2) + 3r + 1] + 2m$	$2n$	n

In this study, it is assumed that the neural network feedforward and backpropagation paths calculations can be carried out within one training time step. However, in practical applications where this assumption can not be easily validated, a tradeoff might seem necessary between the results of the growing/pruning analysis and the computational complexity of the network, which might lead to choosing a smaller number of neurons in the hidden layer.

4.6. Size of the Training Data

After determining the optimum size of the network (according to the criterion in section 3.4), the issue becomes: “How many data samples should be considered so that the neuroidentifier shows acceptable generalization in the testing stage?” Clearly this will

not be an issue in the case of continuous online training, since the network receives new data at every training time step. However, the disadvantage of this technique is that the network might forget the previously learned information if it is continuously trained at a single operating condition.

Vapnik and Chervonenkis (1971) defined a parameter called the *Vapnik-Chervonenkis Dimension* (VCDim) which is a measure of the generalization capability of a system, such as a neural network. The error during the testing stage can be bounded if the number of the training samples exceeds VCDim. Regardless of the number of layers and the type of activation functions used, for a MLP neural network with N_W weights and N_N nodes it can be shown that (Baum and Haussler, 1989):

$$VCDim \leq 2N_W \times \log_e(N_N). \tag{4-2}$$

A useful rule of thumb is that the number of the training samples should be around 10 times the VCDim value (Hush and Horne, 1993). Table 4.2 shows the upper bound for the VCDim of the MLP neuroidentifier.

Table 4.2. VCDim for the RBF neuroidentifier.

ANN architecture	RBF
Input Neurons	7
Hidden Neurons	19
Output Neurons	1
Number of Nodes	27
Number of Weights	19
VCDim Upper Bound	125.24

Using the rule of thumb mentioned above, a training set of 1,250 data samples should be selected for the RBF neuroidentifier. With a sampling time of 20.0 *ms* chosen for the neuroidentifier input data collection, the above data sets require about 25.0 seconds of simulating the power system.

Another important issue regarding the neural network training set is that it should be a true representative of the problem space. This issue is specifically important when it comes to power systems. A multimachine power system will experience various operating conditions due to the changes in the network configuration as a result of the transmission lines and shunt loads being switched on and off, as well as the changes to the power/voltage reference values. Therefore any neuroidentifier should be trained at multiple operating conditions in order to be able to truly model the plant. Its performance should be then tested at a separate operating condition which is not included in the training set. This is necessary in order to evaluate the generalization capabilities of the neuroidentifiers.

Figure 4.5 shows the operating conditions that are used for training (total of 6 points) and testing the neuroidentifier (one point). These points are achieved by changing the power reference of the synchronous generators and the STATCOM line voltage reference, as well as disconnecting one of the transmission lines. Data samples equal to ten times the VCDim of the neural network should be collected at each of the operating conditions. Therefore, each training epoch from now on includes $6 \times 1{,}250$, i.e., 7,500 training samples.

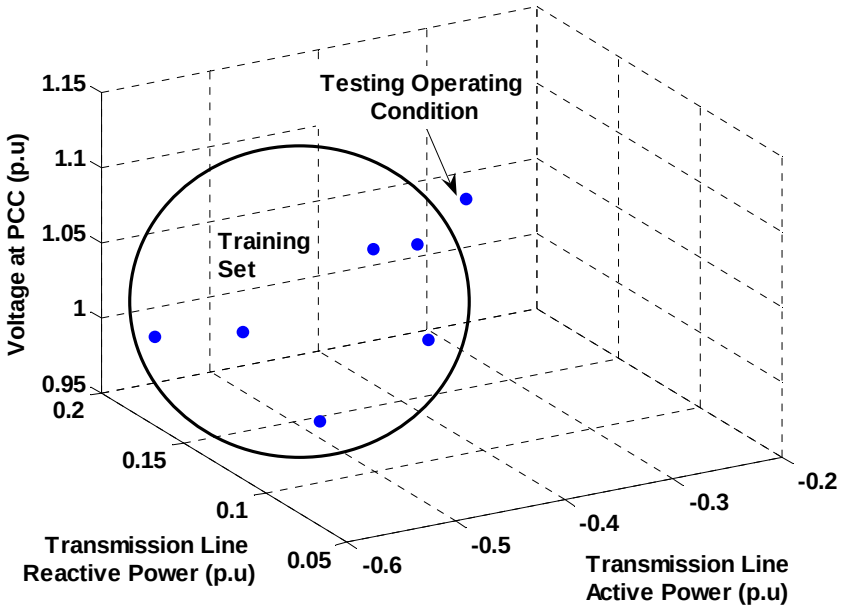

Figure 4.5. Multiple operating points for training the neuroidentifier.

In order for the neuroidentifier to be a true model of the system, every training epoch should include both aspects of system dynamics, namely small signal and large signal dynamics. Small signal dynamics of the power system can be modeled by disturbing the plant input (STATCOM modulation index) from an external source. This disturbance should be small such that it does not change the operating conditions of the system, but cause deviations in the vicinity of that point. Large signal dynamics on the other hand can be modeled by applying severe large scale faults that change the operating point of the system, either permanently (e.g., by switching a big load or a transmission line on/off) or temporarily (e.g., by a short circuit fault that is recovered after a short period of time). This issue has been taken into account for generating data samples at each operating conditions, so that the data in the training epoch is a true representative of the multimachine power system.

4.7. Training the Neuroidentifier

4.7.1. Offline Clustering of RBF Units

The first task in training a RBF neural network is determining the centers. In this study, the centers are determined using an offline clustering method. A data set of 7,500 samples (an epoch) is used for this purpose. The centers are initially assigned to 19 random points in the data set. The clustering algorithm is then iterated on the data set until a certain accuracy level (tolerance) is achieved (see Appendix C). Figure 4.6 shows the results of the simulation after 200 iterations. All the centers are initially concentrated in the region where the largest number of samples exist. However, after the iterations, the centers are dispersed in order to cover all the data points, as shown in Figure 4.6. A close-up view of the most densely populated region in the sample space (Figure 4.6) is provided in Figure 4.7 that illustrates more clearly how most of the randomly assigned initial centers are located at this region, but only a few remain there after the offline training.

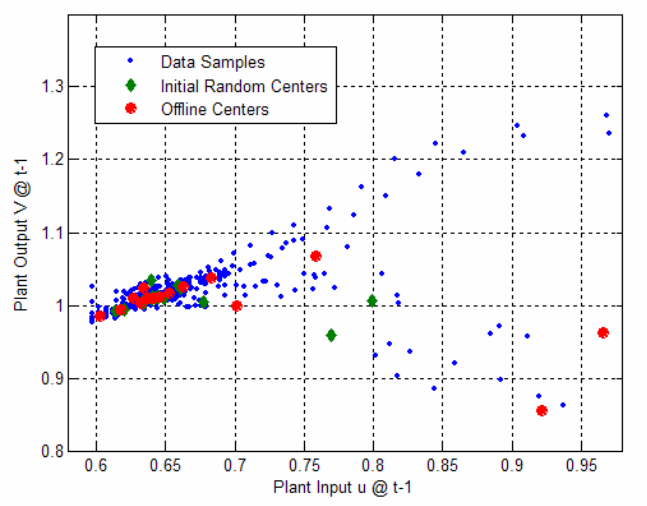

Figure 4.6. Initial and final centers of RBF using offline clustering method.

The centers of each RBF unit are 7-dimensional vectors; however, only the two components, i.e., $V(t-1)$ and $u(t-1)$ are shown in Figures 4.6 and 4.7.

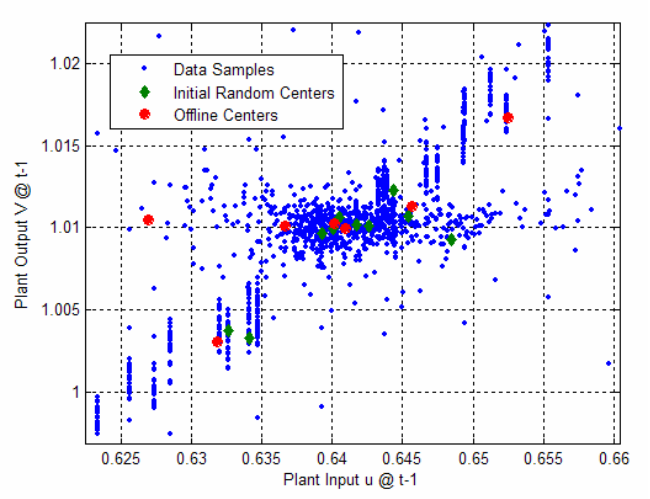

Figure 4.7. Close-up view of the initial and final RBF unit centers.

4.7.2. Forced Training

After determining the RBF centers, the neuroidentifier goes through a forced training stage, in which manual disturbances are applied to the system and the RBF neural network output weight matrix is trained using the backpropagation training algorithm in order to learn the system dynamics (the centers are not updated anymore). This is achieved by applying pseudorandom binary signals (PRBS) generated by an external source to the plant input. The PRBS magnitude is selected in a way that the deviations in the plant output (line voltage at bus 5 in Figure 4.1) are limited to $\pm 5\%$. The PRBS disturbance in this study has a combination of three frequencies 0.5, 1 and 2 Hz, which are close to the natural frequencies of oscillations in the system. Figure 4.8 shows the typical modulation index applied to the plant input, along with the PRBS.

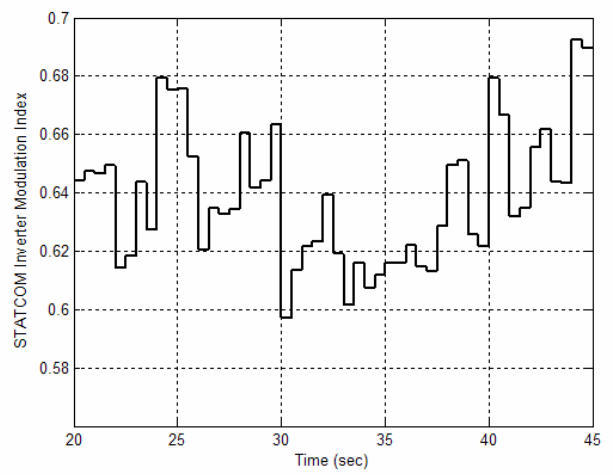

Figure 4.8. PRBS applied to the power system during forced training.

Figure 4.9 shows the block diagram for forced training the neuroidentifier. The system is first run until it reaches the steady state, during which the STATCOM modulation index m_a and the inverter phase shift α become constant numbers m_{a0} and α_0 respectively. At this stage the line voltage PI controller (PI_V) is disconnected (switch S goes from position 1 to position 2) and its constant output m_{a0} is applied to the system from an external source. The PRBS signal is now added to the plant output (switch S in position 3) and training the neuroidentifier starts. The PI_{DC} control loop is not changed during this process.

Figure 4.10 shows the forced training results of the neuroidentifier and illustrates that during this period, the neuroidentifier output follows the plant output with good precision.

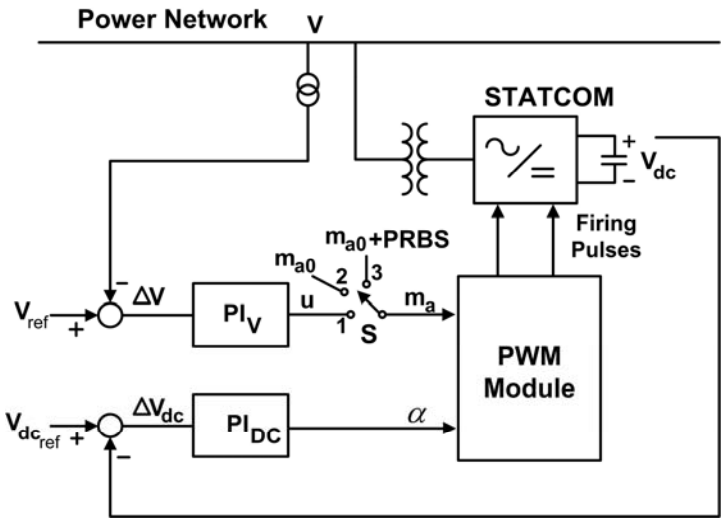

Figure 4.9. Neuroidentifier forced training block diagram.

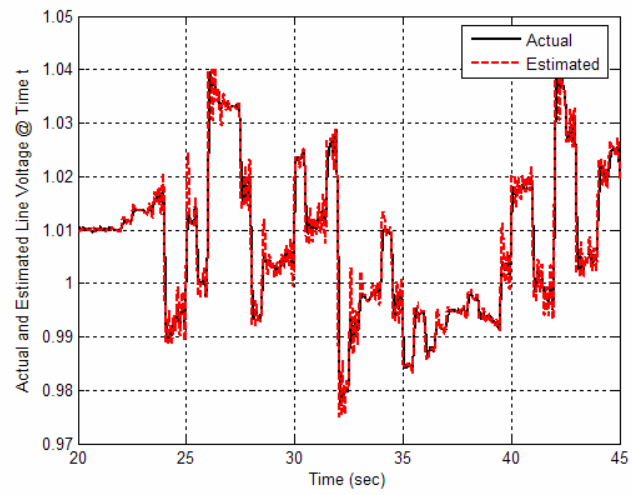

Figure 4.10. Actual and estimated values of the plant output during the forced training stage.

4.7.3. Natural Training

If chosen properly, the PRBS disturbance correctly stimulates the power system's small signal dynamics. However, natural training enables the neuroidentifier to learn the response of the system to large scale disturbances, such as three phase short circuits. During natural training, the RBF centers are held fixed, the PRBS disturbance is removed

and switch S is back in position 1 so that the line voltage PI controller (PI_V) is controlling the plant. The power system is now subjected to various short circuit tests as well as changes to the system configuration, for instance by disconnecting one of the transmission lines or switching on/off a shunt load at bus 5 (Figure 4.1).

4.7.4. Neuroidentifier Training Cross Validation

A neural network that learns a finite set of training samples, yet performs poorly on data samples outside the original training set is said to be overfitting the training samples (Hush and Horne, 1993). Preventing overfitting is one of the important aspects of training a neuroidentifier. While not being an issue in online training, for offline cases overfitting can make the neuroidentifier too perfect for the available training data set, while such a network might perform very poorly for a data set outside the training set. One of the simplest and most widely used means of avoiding overfitting is the cross-validation method, in which the data is divided into two sets: a training set and a testing (validation) set (Figure 4.5). After each epoch of training the neuroidentifier with the training data set, the training stops and the performance of the neuroidentifier is tested by applying a set of testing data and evaluating the MSE. Initially, the testing MSE drops by an increase in the number of training epochs; however, after a while, increasing the number of training epochs increases the testing MSE. This is the point where the neural network starts overfitting the data set.

This test has been carried out repeatedly for the neuroidentifier and the results are shown in Figure 4.11. It can be seen that for different cross-validation tests, overfitting occurs at slightly different training epochs, namely 11^{th} training epoch for test 1, 16^{th}

training epoch for test 2 and 12th training epoch for test 3. Twelve training epochs are selected in this study to ensure no overfitting will happen.

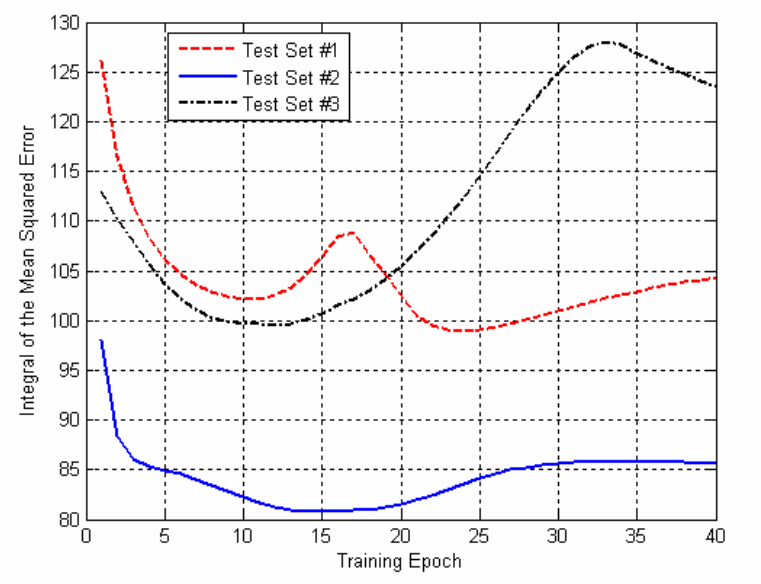

Figure 4.11. Cross-validation of the neuroidentifier.

4.8. Neuroidentifier Testing

With the optimum number of neurons in the hidden layer and the centers determined offline by a clustering method, the neuroidentifier undergoes forced training and natural training stages with the training sets of 1250 samples each, which includes both forced training (PRBS) and natural training samples, as described in the previous sections.

After training the RBF online for 1,800.0 sec (representing 12 training epochs, each one at 6 operating conditions as shown in Figure 4.5), the performance of the neuroidentifier is now being tested at a different operating condition (Figure 4.5) to see how well it can track the plant dynamic.

At the new operating point where the neuroidentifier has not been trained before, PRBS signals are applied to the neuroidentifier for 10 sec, followed by two separate faults: a 100 *ms* three phase short circuit at the terminals of the generator 3, and a 100 *ms* three phase short circuit at the infinite bus. Figures 4.12-4.14 contain the actual and estimated values of the plant output for the above tests and show that the neuroidentifier manages to track the system dynamic with acceptable precision.

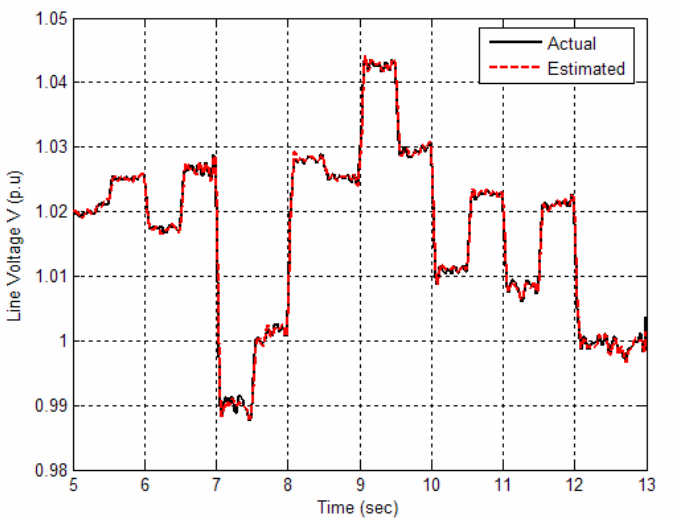

Figure 4.12. Actual and estimated values of the plant output during the testing stage with PRBS.

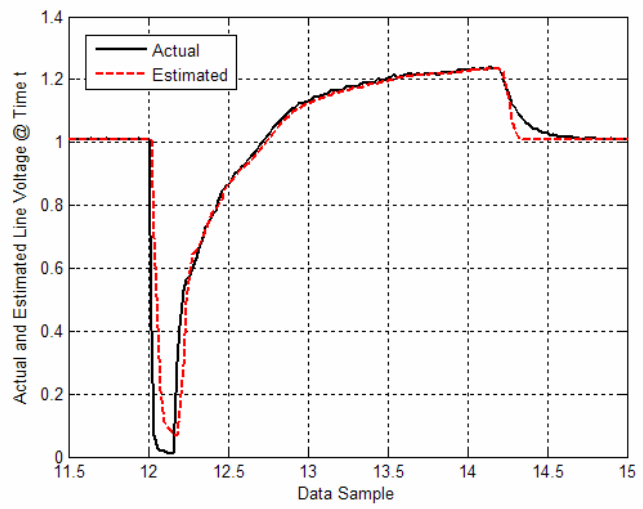

Figure 4.13. Actual and estimated values of the plant output during a 100 *ms* three phase short circuit at the terminals of generator 3.

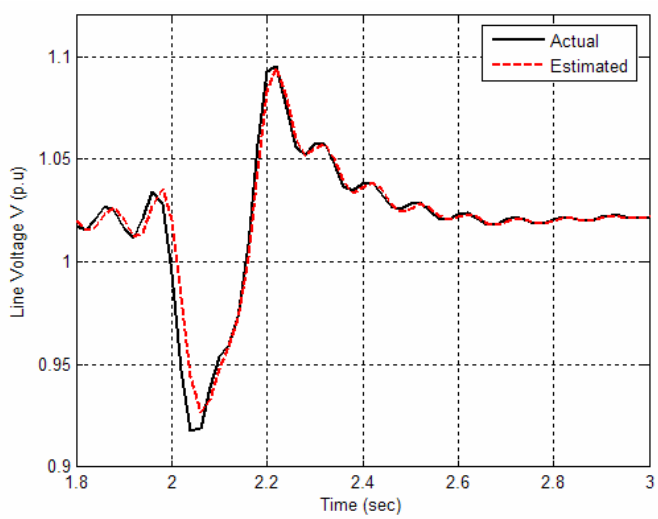

Figure 4.14. Actual and estimated values of the plant output during a 100 *ms* three phase short circuit at the infinite bus.

4.9. Conclusions

This chapter has proposed and implemented a radial basis function neural network identifier that can track the small signal and large signal dynamics of a multimachine power system with a STATCOM. The neuroidentifier does not need any prior knowledge of the plant in the form of mathematical equations. Instead it is designed based on the NARMA model of the system and considers the power system as a black box with an input and an output.

The neuroidentifier is trained using offline data; however, the same procedure applies to online training of the identifier as well. Its performance is then tested by applying various faults and disturbances at an operating point where the neuroidentifier has not been trained before but is able to track the plant output with high precision. Simulation results are provided that indicate the neuroidentifier has learned the system small signal and large signal dynamics.

Such a neuroidentifier can serve as the model of the plant in any model based intelligent control technique implemented for the STATCOM.

CHAPTER 5

STATCOM FUZZY LOGIC BASED CONTROLLER

5.1. Introduction

In this chapter a Takagi-Sugeno (TS) fuzzy logic based controller is designed and implemented in hardware for the STATCOM connected to the multimachine power system in Figure 4.1. The proposed fuzzy controller replaces the line voltage PI controller (PI_V) in Figure 4.1. A real-time digital simulator (RTDS®) is employed that allows the analysis of the power system in real-time. The controller is first designed using fixed parameters and then its performance is further improved by applying an indirect adaptive scheme for updating the control parameters. A neural network based identifier similar to the one explained in Chapter 4 is employed in order to provide the appropriate training signals for the adaptive fuzzy controller. Several scenarios are studied in order to compare the performance of the proposed TS fuzzy controllers with that of a conventional PI_V controller.

5.2. Fuzzy Logic Control

Figure 5.1 shows the schematic block diagram of a typical fuzzy logic based controller. Based on the type of crisp input applied to the controller (error, derivative of the error or the integral of the error) the fuzzy controller can perform as a nonlinear proportional, derivative or integrator controller. In general, fuzzy controllers can be viewed as nonlinear controllers using gain scheduling. They provide a nonlinear mapping from a set of crisp inputs to a set of fuzzy values. These values are processed using fuzzy rules and the resultant fuzzy output is converted to a crisp fuzzy output. A *term set* and a

universe of discourse is associated with every fuzzy variable, also known as a *linguistic variable*. Normally the linguistic variables in a fuzzy system are the state, the state error, the state error derivative or the state error integral (Lee, 1990).

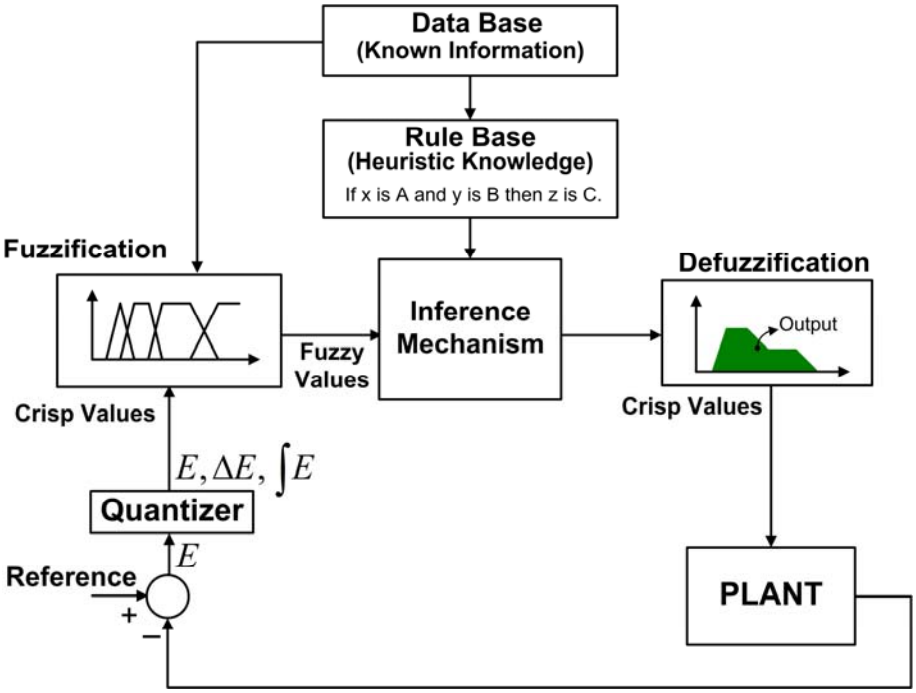

Figure 5.1. Fuzzy logic controller schematic diagram.

Different standard or non-standard functions, called *fuzzifiers*, can be used for mapping crisp values to fuzzy values. Standard fuzzifiers such as the *Singleton*, *Gaussian* and *Triangular* are the most common functions used. However, in general there does not exist a systematic way for selecting the proper membership function for a given problem. Whether standard or non-standard functions are used, they should all be normalized between 0 and 1. These functions are generally overlapping which represents the main concept of the fuzzy controllers: the glass can be partly full and partly empty at the same time.

The heart of any fuzzy system is the knowledge base associated with it. The knowledge base of a fuzzy controller is comprised of two parts: a *data base* and a *rule base*. The data base is the known information about the dynamics and the behavior of the plant, and is obtained from human operator experience and engineering judgment. The rule base is the heuristic information on the performance of the plant and is given as a set of "if - then" rules derived based on the data base.

Based on the data base available on the plant dynamics, the design engineer proceeds as follows:

- Select the appropriate fuzzy variables for controlling the plant,
- Assign the universe of discourse for the fuzzy variables,
- Partition the universe of discourse, i.e., determine how many terms should exist in each term set,
- Assign degrees of membership values to the elements of the universe of discourse, i.e., decide on the form of the fuzzy membership functions.

The rule base of a fuzzy system includes a set of conditional statements (in the form of *Modus Ponens*) (Lee, 1990). The fuzzy rule base can be derived from the data base (expert knowledge/operator experience), mathematical models or a combination of both. It can also be fixed (robust mode) or adaptive (learning mode). The rule base and the subintervals defined for the fuzzy membership functions are closely related to the problem *data base*. Although most fuzzy controllers do not need a mathematical model of the plant to be controlled, some knowledge of the plant, such as its behavior and the range of inputs/outputs, is sometimes necessary for designing an efficient fuzzy rule base. In general, in a multi input single output fuzzy system:

$$\text{Rule } j\text{: } \textbf{If } u_1 \text{ is } F_1^j, \ldots, \text{ and } \textbf{If } u_n \text{ is } F_n^j, \textbf{ Then } y \text{ is } G^j(u_1,...,u_n), \qquad (5\text{-}1)$$

where, depending on the inference mechanism used, the function G^j can be a fuzzy set, a polynomial or a single value. In fuzzy logic, the Modus Ponens rules are extended to *Generalized Modus Ponens*:

$$\begin{aligned} &\textit{Premise:} && u_1 \text{ is } F_1^{j*}, \ldots, \text{ and } u_n \text{ is } F_n^{j*}, \\ &\textit{Implication:} && \textbf{If } u_1 \text{ is } F_1^j, \ldots, \text{ and } \textbf{If } u_n \text{ is } F_n^j, \textbf{ Then } y \text{ is } G^j(u_1,...,u_n), \\ &\textit{Consequence:} && y \text{ is } G^{j*}(u_1,...,u_n). \end{aligned} \qquad (5\text{-}2)$$

In crisp logic a rule will be fired only if the premise is exactly the same as the antecedent of the rule. In fuzzy logic on the other hand, a rule is fired as long as there is a non-zero degree of similarity between the premise and the antecedent of the rule (Mendel, 2001).

The fuzzy inference mechanism, also referred to as *fuzzy model*, applies the fuzzy reasoning to the rules in the rule base in order to derive a mathematically reasonable output or conclusion which represents the problem conditions best. Three types of the more frequently used fuzzy inference systems are *Mamdani*, *Takagi-Sugeno* and *Tsukamoto* fuzzy models (Jang and Sun, 1995); among which the Mamdani method and the Takagi-Sugeno method are the most common methods for designing fuzzy controllers (Ding *et al.*, 2003; Lee, 1990). These are universal approximators of any continuous function to any degree of accuracy. Generally speaking, more fuzzy sets and rules in the rule base are needed to gain better approximation accuracy (Ying *et al.*, 1999). Some of the advantages of both methods are as follows (Mendel, 2001; Passino and Yurkovic, 1997): The Mamdani method is intuitive, has widespread acceptance and is well suited to human input, while the Takagi-Sugeno method is computationally efficient, works well

with linear, adaptive and optimization techniques, has guaranteed continuity of the output surface, and is well suited to mathematical analysis.

In short, the Mamdani method is the better option for static systems (with slow changing dynamics) and the Takagi-Sugeno method is more efficient for dynamic systems (with fast changing dynamics).

Various rules in the rule base are applied to the fuzzy input data, in order to create a fuzzy output according to the selected inference mechanism. The resultant fuzzy output should now be converted to a crisp value in order to be applied to the plant. Various defuzzification techniques exist that can be used for this purpose, namely the *Max* method, *Mean of Maximum* (MOM) method and the *Center of Area* (COA) method. In general, a defuzzifier is a mapping that, given a fuzzy set, determines the best crisp representative of that set.

The MOM strategy has a better transient performance, while the COA method provides a better steady state performance. In general, COA yields a lower mean squared error than the MOM method, while both of them are more efficient than the Max criterion (Lee, 1990). The COA method, also known as *centroid* defuzzifier, is used in this study as the mapping technique.

5.3. STATCOM Fuzzy Logic Controller

5.3.1. Controller Structure

The fuzzy controller designed in this study, replaces the line voltage PI controller of the STATCOM (controller PI_V in Figure 5.1). The second PI controller in Figure 5.1 (dc link voltage controller PI_{DC}) is not replaced by a fuzzy logic controller. This second PI controller is able to maintain the capacitor voltage within defined limits and unlike the

power network, the STATCOM topology does not change. The fuzzy controller has two inputs, the line voltage deviation $\Delta V(t)$ and the change in the line voltage error $\Delta E(t)$, i.e. $\Delta V(t) - \Delta V(t-1)$. Using the line voltage error helps the controller to respond faster and more accurately to disturbances in the system. A time step of 2.0 *ms* is selected for calculating the change in error. In turn, the controller generates a control signal to the plant, i.e. $\Delta u(t)$. Figure 5.2 shows the schematic diagram of the proposed fuzzy controller.

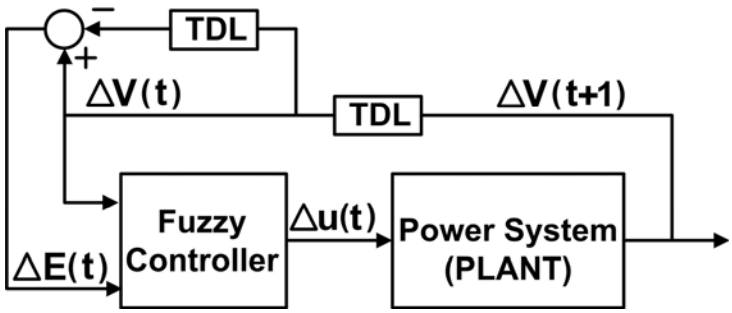

Figure 5.2. Fuzzy logic controller structure.

The controller in Figure 5.2 can only work with the deviation signals. The system in Figure 5.1 is first run with the PI controllers in the circuit, until it reaches steady state, when the PI_V and controller output is a constant value u_0. The PI_V is now deactivated and its constant output is applied to the plant by moving switch S in Figure 5.1 is moved from position 1 to position 2. The fuzzy controller output $\Delta u(t)$ is then added to this constant value in order to form the final control input to the plant, i.e., $u_0 + \Delta u(t)$.

A proportional-integrator approach can also be applied in order to enable the fuzzy controller to deal with the actual signals rather than deviation signals. This is achieved by

adding the instantaneous controller output $\Delta u(t)$ to the previously accumulated total control signal (Figure 5.3). The final control output:

$$u(t) = u(t-1) + \Delta u(t), \tag{5-3}$$

replaces the final control output $u(t)$ replaces the inverter modulation index $m_a(t)$ in Figure 5.1. It can be proven that such a TS fuzzy controller is inherently a nonlinear gain scheduling PI controller (Mendel and Mouzouris, 1997).

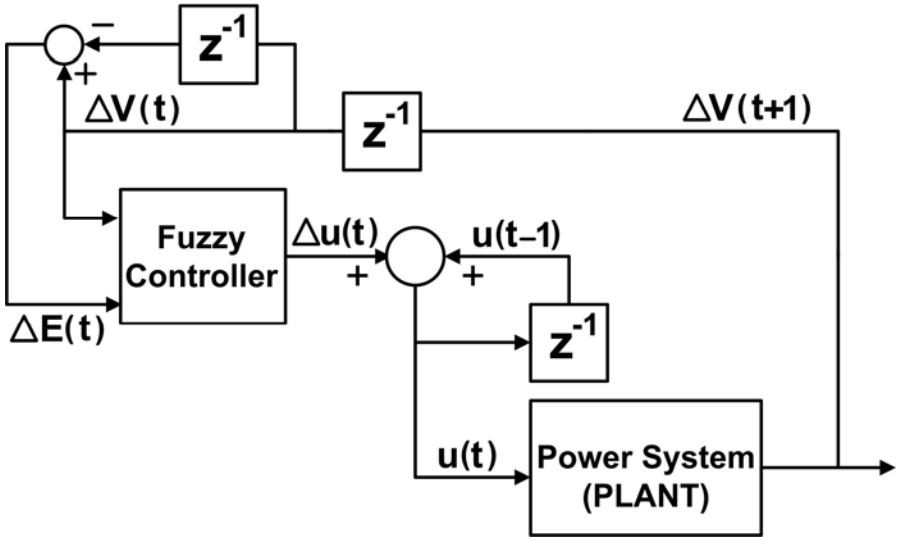

Figure 5.3. Proportional-integrator fuzzy controller structure.

5.3.2. Shrinking Span Membership Functions (SSMF)

There are several design parameters associated with a fuzzy logic controller, each of which plays an important role in the efficiency of the controller. Selecting the proper fuzzy membership functions, subintervals and the defuzzification method, are some of them. Due to simplicity, most researchers tend to design the input/output fuzzy membership functions using so called equal-span functions. However, these functions do not necessarily provide the optimum solution for all problems. Instead a prior knowledge

of the plant to be controlled, and its dynamics, might lead to different standard or non-standard fuzzy membership functions with various physical shapes. Moreover, when the control response is closer to the system set point, it can be intuitively seen that the fuzzy membership functions for that specific linguistic term should have narrower spans, in order to be able to provide smoother results with less oscillations.

Shrinking span membership functions are proposed by Chen and Hsieh (1996) in order to compensate for the above problems. This method creates triangular membership functions with shrinking spans (Figure 5.4), in a way that the controller generates large and fast control actions when the system output is far from the set point and makes moderate and slow changes when it is near the set point. Mohagheghi *et al.* (2004) compared the efficiency of the SSMF method with the conventional membership function design for a fuzzy logic controller for a STATCOM in a multimachine power system, and showed that the SSMF method is more efficient in controlling the STATCOM.

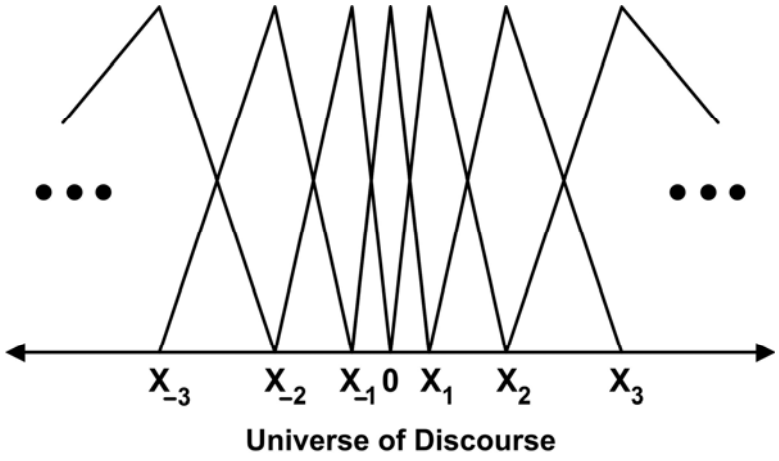

Figure 5.4. Shrinking span membership functions.

The details of designing a SSMF fuzzy controller in a general case (multiple input multiple output systems) is rigorously described in (Chen and Hsieh, 1996). Nevertheless, it is briefly revisited here for this specific problem (single-input-single-output system). Different triangular functions for the input variable can be expressed as in Equation (5-4):

$$F_i(x) = \Delta(x; x_{i-1}, x_i, x_{i+1}), \quad \text{for } i = -m, ..., m \tag{5-4}$$

where m is the index for the input set, resulting in $2m+1$ linguistic terms for the input variable x. The function Δ in Equation (5-4) is a triangular function defined as:

$$\Delta(x; a, b, c) = \max[0, \min(\frac{x-a}{b-a}, \frac{d-x}{d-c})], \tag{5-5}$$

and the subintervals x_i's are derived as:

$$x_i = \frac{i}{m} \times s^{m-|i|}, \tag{5-5}$$

where $s \in [0,1]$ is the *shrinking factor* for the input variable x. A typical shrinking factor of 0.7 is selected in this study. With the mathematical definition of the membership functions in Equations (5-4) and (5-5), the universe of discourse is limited to [-1,1], however this can be changed by introducing a scaling factor.

5.3.3. Rule Base

In this study, the parameter m is selected to be 3 for the line voltage deviations and the controller output, and *1* for the line voltage error. Therefore 7 shrinking span membership functions are assigned to the line voltage deviation $\Delta V(t)$ and the controller output $\Delta u(t)$, namely *Negative Big*, *Negative Medium*, *Negative Small*, *Zero*, *Positive Small*, *Positive Medium* and *Positive Big*. Three functions, *Negative*, *Zero* and *Positive*, are also considered for $\Delta E(t)$. These rules are summarized in Table 5.1.

Table 5.1. Fuzzy controller rule base.

Fuzzy Inputs/Output		ΔV						
		NB	**NM**	**NS**	**Z**	**PS**	**PM**	**PB**
ΔE	**N**	NB	NB	NM	Z	PS	PM	PB
	Z	NB	NM	NM	Z	PM	PM	PB
	P	NB	NM	NS	Z	PM	PB	PB

5.3.4. Takagi-Sugeno Model

Due to the fact that a STATCOM (or any other FACTS device) in a power system goes through fast changes in terms of system parameters and dynamics, the Takagi-Sugeno (TS) model is selected for designing the STATCOM fuzzy logic controller. The TS model was proposed by Takagi, Sugeno and Kang in an effort to develop a systematic approach for generating fuzzy rules from a given data set (Takagi and Sugeno, 1985; Sugeno and Kang, 1988). A fuzzy rule in the Takagi-Sugeno inference mechanism can be typically expressed as:

$$\text{Rule } j\text{: } \textbf{If } u_1 \text{ is } F_1^j, \ldots, \text{ and } \textbf{If } u_n \text{ is } F_n^j, \textbf{ Then } y = f_j(u_1,...,u_n), \tag{5-6}$$

where the antecedent values $F_1^j, \ldots, F_n^j$ are fuzzy sets and the function f_j in the consequent is a crisp function, usually a polynomial. Depending on the order of the fuzzy consequent function f_j, the controller can be a zero, first or higher order TS model. The zero-order TS model can also be viewed as a special case of the Mamdani fuzzy inference system, in which each rule's consequent is specified by a fuzzy singleton.

Since each rule has a crisp output, the overall output of the fuzzy controller is obtained using the centroid defuzzification in order to provide a smooth result. The instantaneous output of the controller can be written as:

$$\Delta u(t) = \frac{\sum_{j=-m}^{m} w_j . f_j(x,t)}{\sum_{j=-m}^{m} w_j}, \tag{5-7}$$

where w_j is the rule firing strength and $f_j(.)$ is the typical linear function of the input variables as in Equation (5-8):

$$f_j(\Delta V(t), \Delta E(t)) = \beta_{j1} \times \Delta V(t) + \beta_{j2} \times \Delta E(t) + \beta_{j3}, \tag{5-8}$$

where in this study the β_{j1} coefficients are considered the only non-zero coefficients.

The coefficients of the consequent functions are derived by trial and error, by testing the STATCOM performance during the step change disturbances as well as large scale faults.

5.4. Indirect Adaptive Fuzzy Controller

Designing a fuzzy controller requires prior information of the plant. This information need not be in the form of a mathematical model, but it should reflect the nature of the system response to various control signals. More accurate information leads to a better and more robust performance of the fuzzy controller. However, even information available on the behavior of the plant does not necessarily lead to optimal fuzzy controller parameters. While a fuzzy controller can be more robust and efficient than the conventional PI controller, it may still suffer from the disadvantage of having fixed parameters. This can *slightly* degrade its performance over the range of the system operating conditions. The performance of the fuzzy controller can further be improved by applying an adaptive technique for updating the parameters of the controller. An *Indirect Adaptive* technique can be applied in which outputs of the plant to be controlled are estimated using a neuroidentifier and the controller parameters are adjusted based on

these estimates (Narendra and Parthasarathy, 1990). While this technique proves to be efficient for the ever changing nature of the power system, as opposed to most common adaptive techniques, it has the advantage of not needing a mathematical model of the plant to be controlled. Instead, a neural network can be trained online to learn the dynamics of the power system (Chapter 4).

The structure of the proposed indirect adaptive fuzzy controller is depicted in Figure 5.5. The neuroidentifier shown in Figure 5.5 is used in order to predict the plant output one time step ahead, at $(t+1)$, and its RBF structure is discussed in Chapter 4. The only difference is that in this specific application, the neuroidentifier is trained to learn the plant input/output deviations rather than the actual signals. This is due to the fact that the proposed fuzzy controller is designed to work with deviation signals.

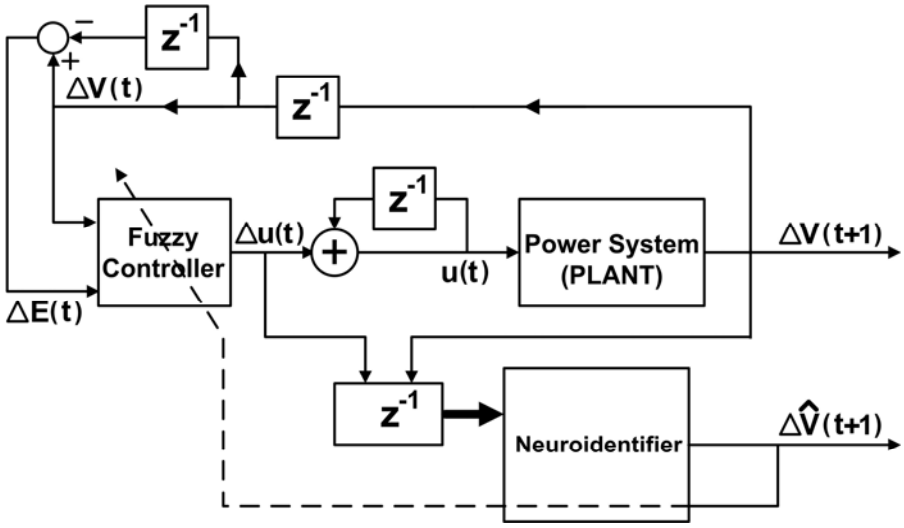

Figure 5.5. Indirect adaptive fuzzy controller structure.

Since the neuroidentifier predicts the output voltage deviations, its output can be considered as the predicted output error at time $(t+1)$, i.e., $\Delta\hat{V}(t+1)$. This value is

compared with the desired response of the system, which in this case is zero, in order to form an error signal. The cost function J is defined as:

$$J(t) = \frac{1}{2}.[\Delta \hat{V}(t+1)]^2, \tag{5-9}$$

The coefficients of the fuzzy output polynomials $f_j(.)$ are the adaptive parameters of the fuzzy controller. A gradient descent learning algorithm is applied for adjusting these values, where each parameter β is updated in the negative direction of the gradient of the cost function J:

$$\beta_{j1}(t+1) = \beta_{j1}(t) - \eta \frac{\partial J(t)}{\partial \beta_{j1}(t)}, \tag{5-10}$$

where η is the learning rate which is considered to be 0.005 in this study. The partial derivative of the cost function with respect to any parameter β_j can be derived using the chain rule:

$$\frac{\partial J(t)}{\partial \beta_{j1}(t)} = \frac{\partial J(t)}{\partial \Delta \hat{V}(t+1)} \times \frac{\partial \Delta \hat{V}(t+1)}{\partial u(t)} \times \frac{\partial u(t)}{\partial \beta_{j1}(t)}. \tag{5-11}$$

The first term in Equation (5-11) is equal to $\Delta \hat{V}(t+1)$ and the second term can be derived by backpropagating the constant value of 1.0 through the neuroidentifier (see Appendix D). The reader is referred to (Feldman, 1996) for mathematical proofs. The last term in Equation (5-11) can also be simplified as follows:

$$\frac{\partial u(t)}{\partial \beta_{j1}(t)} = \frac{w_j \times \Delta V(t)}{\sum_{j=-m}^{m} w_j}. \tag{5-12}$$

A *partial updating* scheme is used for this study in which only the coefficients β_j of the fuzzy controller output polynomials are updated. It is also possible to apply a *full updating* scheme where the parameters of the membership functions are adaptively

adjusted as well. The same logic mentioned above can be extended for implementing the full updating scheme.

The adaptive fuzzy controller should be trained by the cost function defined in Equation (5-9) using the update equations in Equations (5-10) and (5-11), so that its output coefficients are adjusted for optimum performance. This process is called *forced training* and is achieved by applying pseudorandom binary signal (PRBS) disturbances to the voltage reference of the plant from external sources (Switch *S* in Figure 5.6 is in position 1). These disturbances are limited to $\pm 5\%$ of the rated voltage reference. At this stage, the fuzzy controller is controlling the plant (Figure 5.6).

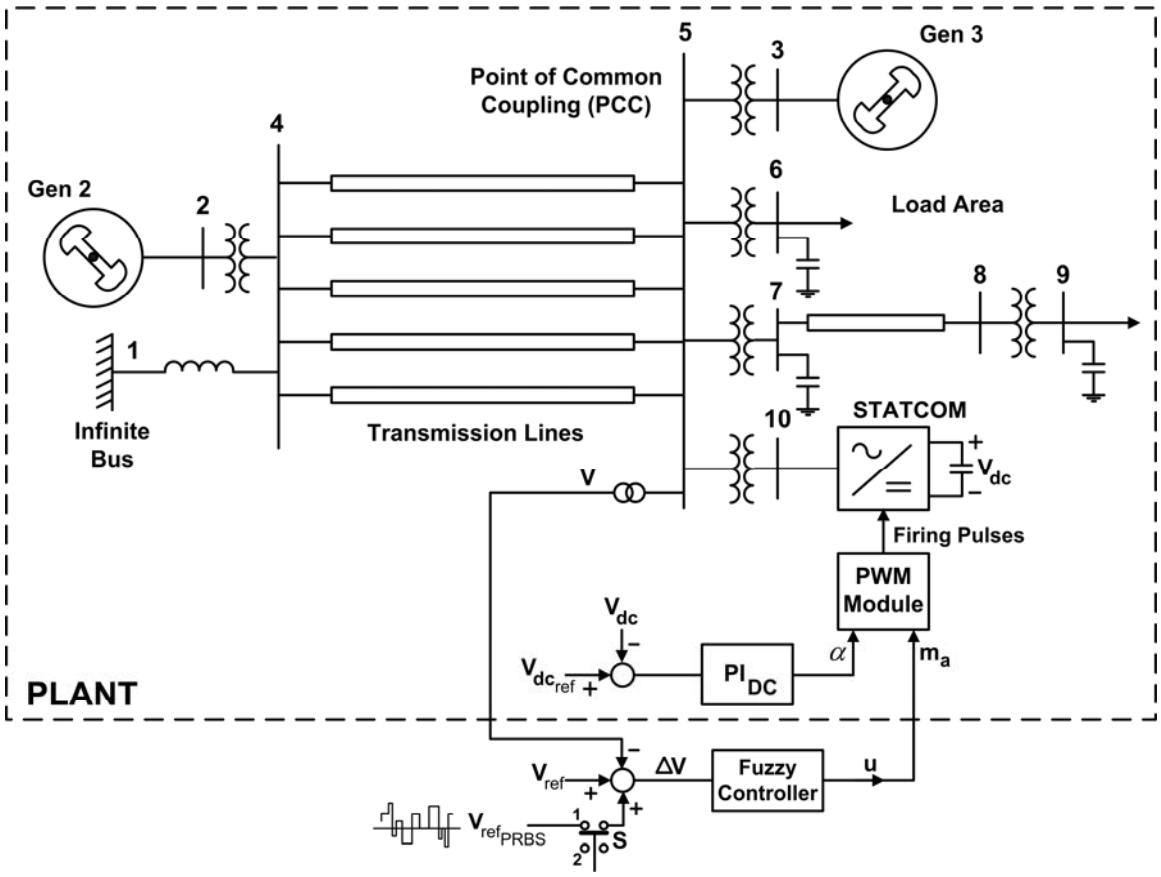

Figure 5.6. Fuzzy logic controller in the multimachine power system.

Typical results of the forced training stage of the fuzzy controller are depicted in Figure 5.7. The training process continues until no further improvement is observed in the performance of the fuzzy controller. At this point the PRBS signal is terminated and the controller is now tested with likely natural disturbances applied to the system (as in Section 4.7.3), as the training continues. However, the learning rate η is reduced from its original value to 0.001 so that it prevents the fuzzy controller output coefficients from sudden changes in response to severe disturbances to the network. An alternative solution for training during large scale disturbances is to apply an adaptive learning gain which automatically decreases when the value of the error signal increases. The former is used in this study.

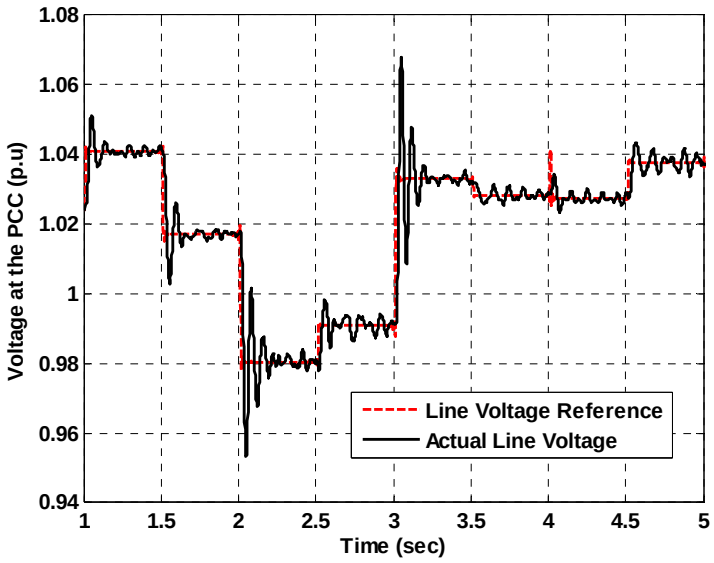

Figure 5.7. Voltage at the PCC (Figure 5.6) during the forced training stage of the fuzzy controller.

5.5. Simulation Results

5.5.1. Case Study 1: Step Change to the STATCOM Voltage Reference

The system in Figure 5.6 is first simulated until it reaches steady state. The PRBS is removed by moving the switch *S* in Figure 5.6 from position 1 to position 2. At 1 second, a 5% step change is applied to the line voltage reference of the STATCOM. The performances of the three controllers are depicted in Figure 5.8, which shows that the adaptive fuzzy controller reaches the target the fastest with no considerable overshoot, as if the system is critically damped. The PI_V controller is fine tuned at a certain operating point as described in Section 2.6.

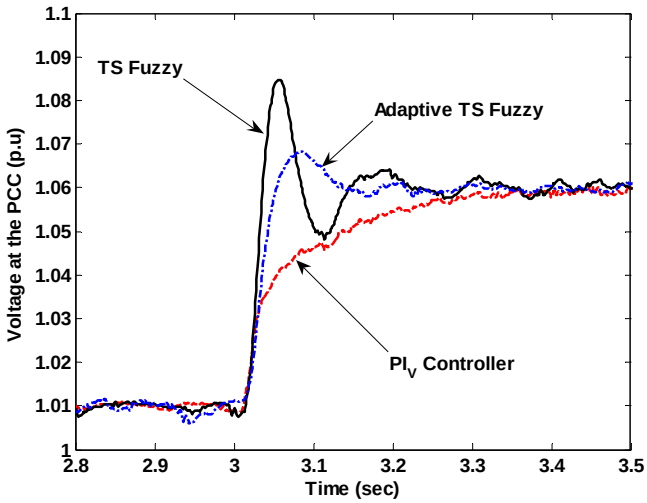

Figure 5.8. Voltage at bus 5 (Figure 5.6) during a step change in the STATCOM line voltage reference.

5.5.2. Case Study 2: Short Circuit at the Terminals of Generator 3

In another test, a 100 *ms* three phase short circuit is applied at the terminals of generator 3 in Figure 5.6. The generator is disconnected from the network and 50 *ms* after the fault is cleared, the generator is switched back on to the system. Figure 5.9 shows the

voltage at bus 5 during and after the fault. The PI_V controller has a poor performance that results in a considerable voltage swell. Such overvoltage can damage sensitive devices connected to the network and/or cause insulation stress. However, the two fuzzy controllers manage to damp out the oscillations quickly. Figure 5.10 shows a close-up view of the period immediately after the fault is cleared, where it can be seen that the adaptive fuzzy performs slightly better than the normal fuzzy.

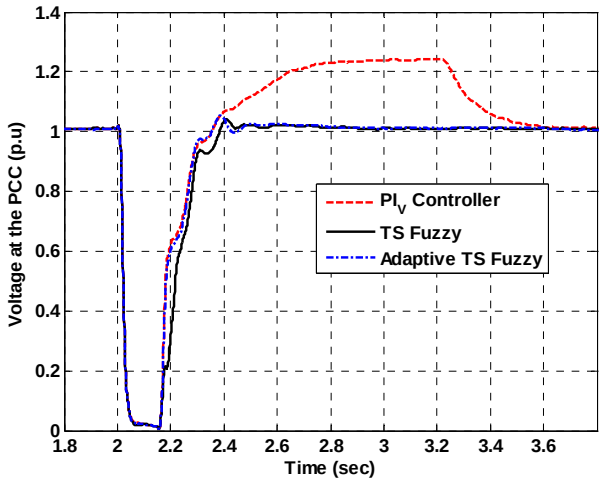

Figure 5.9. Voltage at bus 5 (Figure 5.6) during case study 2.

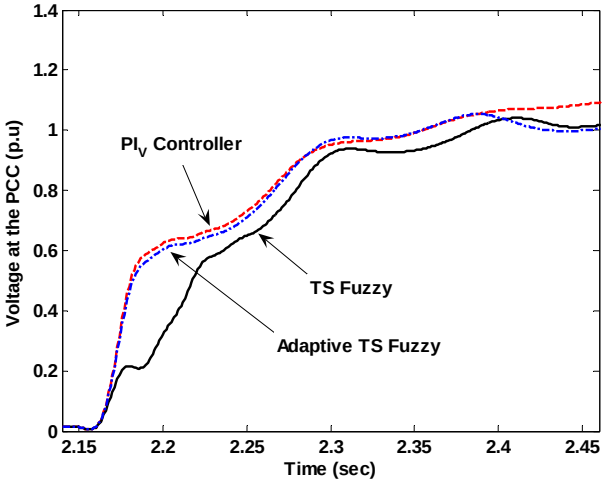

Figure 5.10. Close-up view of voltage at bus 5 (Figure 5.6) during case study 2.

The controllers can also be compared in terms of the control effort provided by each one's STATCOM to deal with the fault. It is shown in Figure 5.11 that the two fuzzy controllers inject less amounts of reactive power into the network compared to the PI_V controller. A large amount of reactive power injected means larger currents pass through the inverter switches, which in turn requires that the ratings of the switches should be increased.

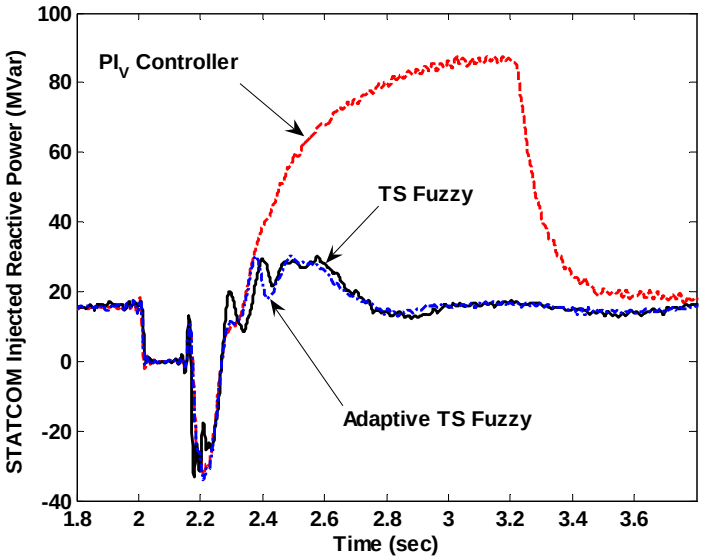

Figure 5.11. Reactive power injected by the STATCOM during case study 2.

5.5.3. Case Study 3: Short Circuit at the Terminals of Generator 3 with Anti-Windup Limiter for the PI_V Controller

The simulation results in the previous section showed that not only does the PI_V controller cause a considerable overvoltage compared to the fuzzy controllers, but it also forces the STATCOM to inject a much larger amount of reactive power into the power system. In those results the modulation index of the STATCOM inverter is clamped at

1.0 by a hard limiter in order to avoid over-modulation. Setting this limit to a lower value helps reduce the peak overvoltage at the PCC as well as the reactive power injection by the STATCOM. Limiting the STATCOM reactive power injection is common in most real world applications, where the STATCOM size is limited.

Figure 5.12 illustrates the voltage at the PCC, when the hard limiter clamps the modulation index of the STATCOM inverter at 0.78. This ensures that the amount of reactive power injected by the STATCOM does not exceed 40 MVar, which is half the value required in Figure 5.11. The code for the fuzzy controller has also been modified to limit the fuzzy controller output to 0.78. The same test as in case study 2 is applied to the power system. Figure 5.12 shows that with the PI_V hard limiter set at 0.78, the peak value of the voltage at the PCC is around 1.1 p.u, which is less than the peak value in Figure 5.9.

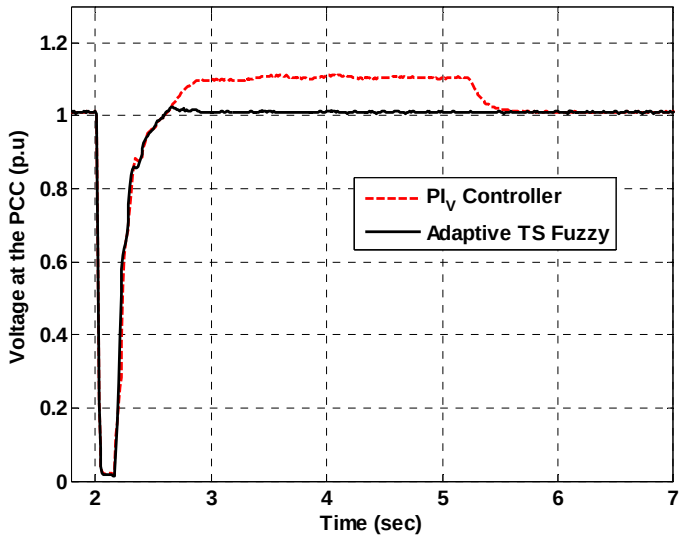

Figure 5.12. Voltage at bus 5 (Figure 5.6) during case study 3, with a hard limiter for the PI_V controller set at 0.78.

Also, Figure 5.13 illustrates the amount of reactive power injected to the power system, which is now limited to 40 MVar.

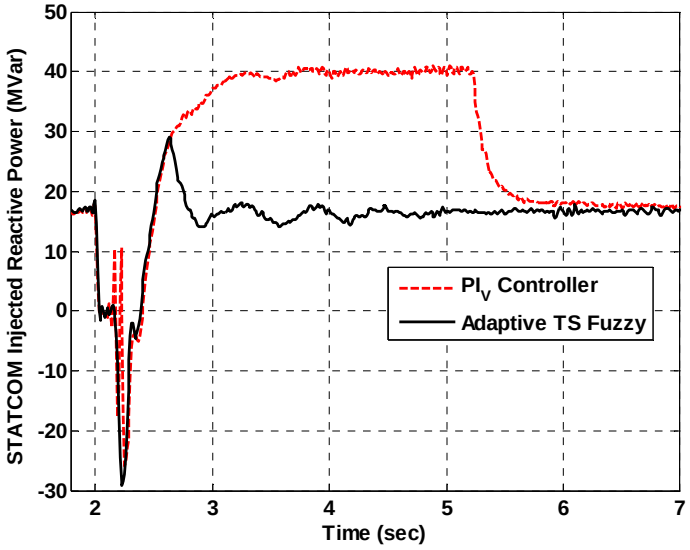

Figure 5.13. Reactive power injected by the STATCOM during case study 3, with a hard limiter for the PI_V controller set at 0.78.

However, comparing Figures 5.9 and 5.12 clearly shows that reducing the threshold of the PI_V hard limiter increases the overall error of this controller. This is due to the fact that during the time when the controller output reaches the limit and is clamped by the limiter, the controller is temporarily working in an open loop fashion, since the feedback from the power system does not impact the controller output. As a result of this, the instantaneous error increases and the PI controller continues accumulating this error. This causes the PI controller to restore the system to the steady state conditions in a much longer time (4 seconds in Figure 5.12 compared to 1.6 seconds in Figure 5.9).

One way to overcome this problem is to use an anti-windup limiter scheme as shown in Figure 5.14 (Åström and Hägglund, 1995). It compares the control signal right at the output of the PI controller with the signal at the output of a hard limiter. The upper and lower limits of the hard limiter in Figure 5.14 are chosen in such a way that improves the performance of the PI controller during large scale disturbances. The difference of the two signals is now fed back to the integrator and is subtracted from the signal going through the integrator. This reduces the instantaneous input to the k_i block and therefore improves the performance of the PI controller. The controller parameters k_P, k_i and k_a are fine tuned at a single operating condition (details appear in Appendix F).

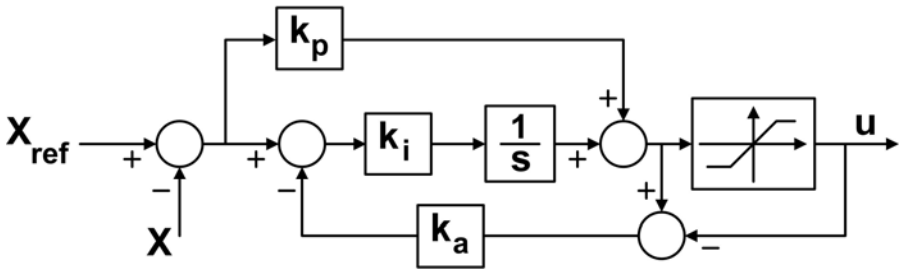

Figure 5.14. Schematic diagram of a PI controller with an anti-windup limiter.

Figure 5.15 shows the response of the adaptive TS fuzzy controller as well as the PI_V controller with anti-windup limiter. The performance of the PI_V controller is drastically improved by applying the anti-windup limiter scheme of Figure 5.14. However, the adaptive fuzzy controller still manages to restore the power system to steady state with less overshoot. Also, Figure 5.16 illustrates the reactive power injected by the STATCOM during the fault. With the anti-windup limiter, the maximum reactive power injection is limited to 45 MVar.

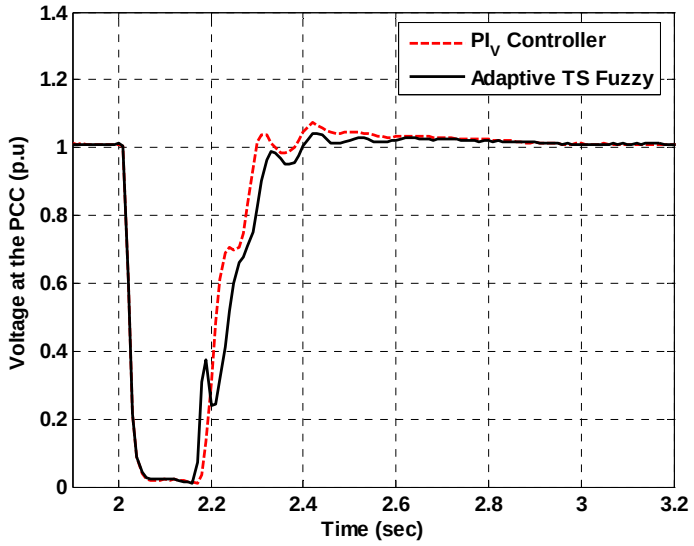

Figure 5.15. Voltage at bus 5 (Figure 5.6) during case study 3, with an anti-windup limiter for the PI_V controller set at 0.80.

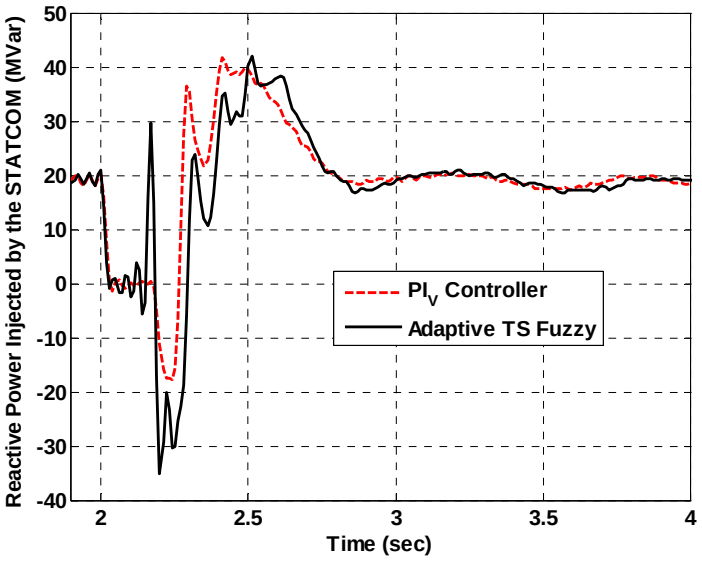

Figure 5.16. Reactive power injected by the STATCOM during case study 3, with an anti-windup limiter for the PI_V controller set at 0.80.

5.5.4. Case Study 4: Short Circuit at a Transmission Line with Anti-Windup Limiter for the PI_V Controller

In this section a 100 *ms* three phase short circuit is applied at the middle of one of the parallel transmission lines in Figure 5.6. The line is removed after the fault is cleared. The PI_V controller is equipped with an anti-windup limiter that clamps the inverter modulation index at 0.8.

Figure 5.17 illustrates the voltage at the PCC. The fuzzy controller manages to restore the system to steady state condition with a temporary peak voltage of 1.06 p.u, whereas the PI_V controller forces the voltage at the PCC to go up to 1.12 p.u.

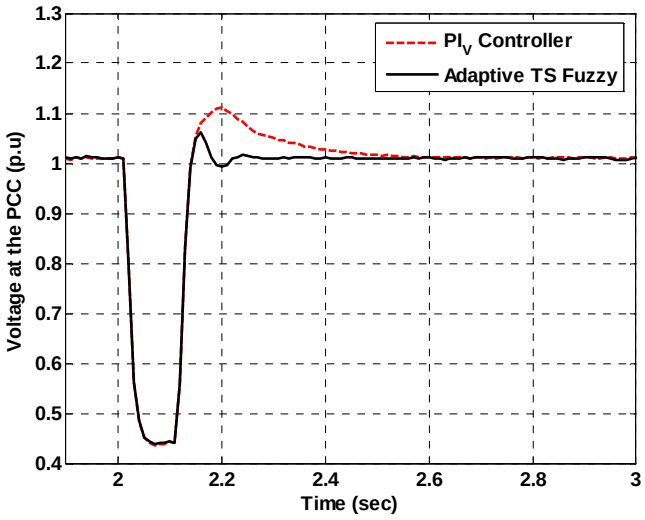

Figure 5.17. Voltage at bus 5 (Figure 5.6) during case study 4, with an anti-windup limiter for the PI_V controller set at 0.80.

Figure 5.18 shows the reactive power injected by the two controllers. Also, the modulation index of the STATCOM inverter is illustrated in Figure 5.19.

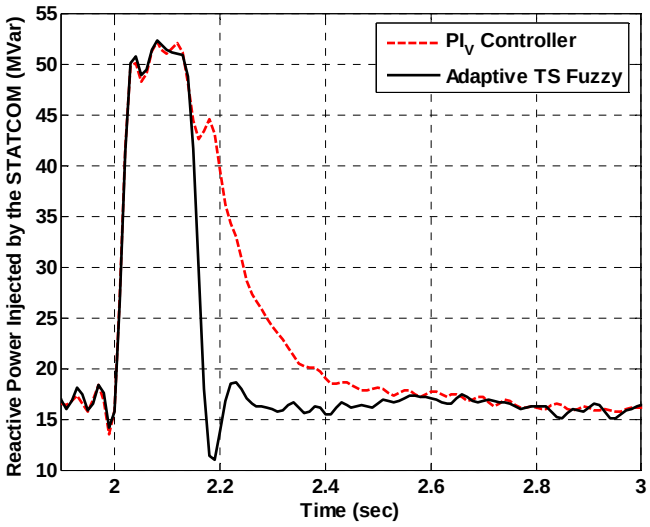

Figure 5.18. Reactive power injected by the STATCOM during case study 4, with an anti-windup limiter for the PI_V controller set at 0.80.

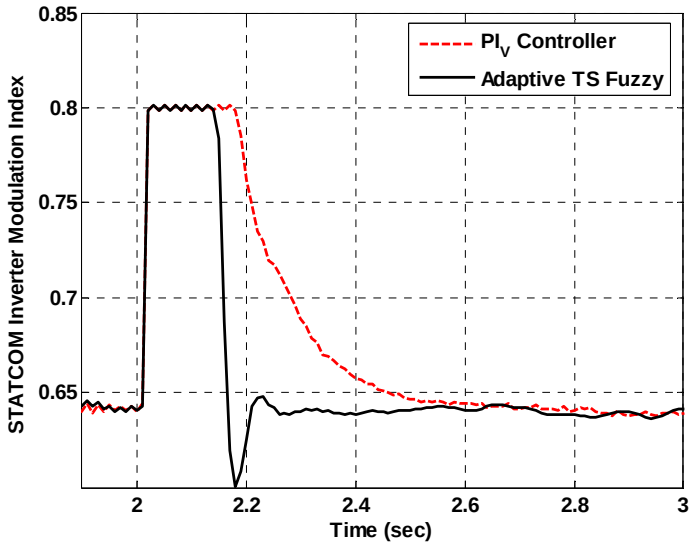

Figure 5.19. STATCOM inverter modulation index during case study 4, with an anti-windup limiter for the PI_V controller set at 0.80.

Figure 5.20 shows the degrees of membership for the fuzzy controller output during the fault. The output functions f_j in Equation (5-8) are fired proportional to the degrees of membership of their corresponding membership functions as in Equation (5-7).

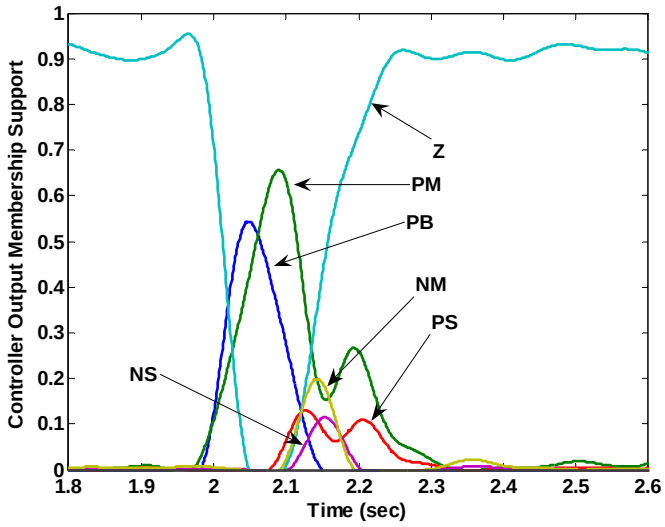

Figure 5.20. Degrees of membership of the fuzzy controller output during case study 4.

5.6. Hardware Implementation

The proposed fuzzy controller in Figure 5.2 is implemented and evaluated in a laboratory setup (Figure 5.21) which consists of a real-time power system simulator and a fuzzy logic controller board.

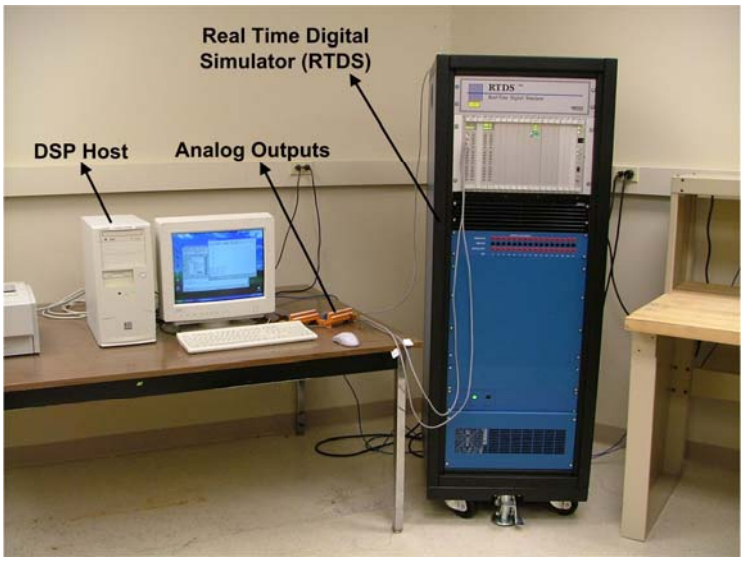

Figure 5.21. Laboratory setup.

5.6.1. Real Time Digital Simulator (RTDS®)

The RTDS® is a fully digital electromagnetic transients power system simulator that operates in real time. It has a custom parallel processing hardware architecture assembled in modular units called racks. Power system equipment and network designs can be evaluated and accurately tested on the RTDS®. Due to the fact that the RTDS® simulator works in continuous, sustained real time, it can solve the power system equations fast enough to continuously produce output conditions that realistically represent conditions in the real network. Because the solution is in real time, the simulator can be connected directly to power system control equipment (RTDS, 1995), such as a fuzzy logic controller.

The RTDS® software is divided into two main categories: the graphical user interface and the underlying solution algorithms for network equations and component models. All aspects of simulator operation, from constructing simulation circuits to recording simulation results are controlled through the graphical interface, the RSCAD software suite, as the user's main interface with the RTDS® hardware.

There are two main RSCAD modules: the Draft and the RunTime. The Draft software module is used for circuit assembly and parameter entry. The RunTime software module is used to control the operation of the RTDS® simulator. Through the RunTime module, the user performs actions such as starting and stopping the simulation cases, initiating system disturbances, changing system set-points, online monitoring of system quantities and suchlike.

The multimachine power system with the STATCOM in Figure 4.1 is modeled on the RTDS® in the RSCAD environment.

5.6.2. Fuzzy Logic Controller

The Mamdani fuzzy logic controller developed in Section 5.3.1 is implemented on an Innovative Integration M67 DSP card (Innovative Integration, 2000) based on the TMS320C6701 digital signal processor operating at 160 MHz, hosted on a Pentium III 433 MHz personal computer. The M67 DSP card is equipped with A4D4 OMNIBUS A/D and D/A conversion modules (Innovative Integration, 2001).

Each A4D4 OMNIBUS module provides the DSP card's processor with four channels of high speed 200 kHz 16-bit resolution A/D output conversion per module slot, as well as four channels of high speed 200 kHz 16-bit resolution D/A conversion. Figure 5.22 illustrates the block diagram of the laboratory setup.

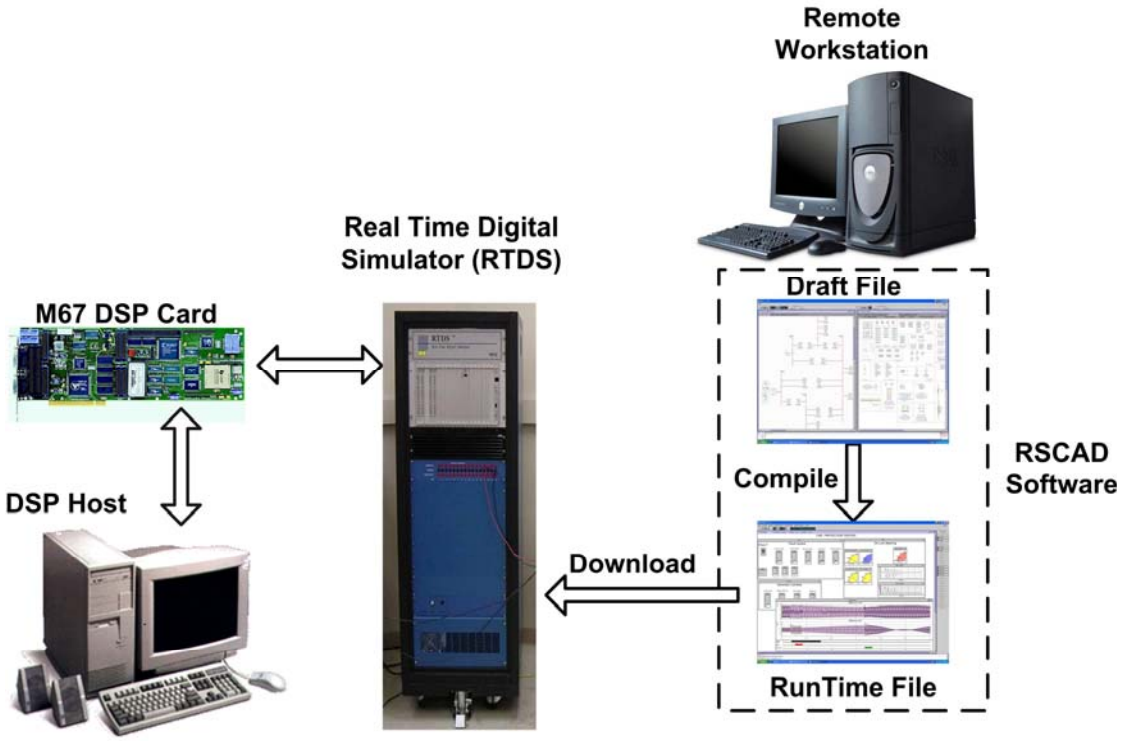

Figure 5.22. Experimental setup block diagram.

5.7. Experimental Results

The conventional PI_V controller is fine tuned at one operating point (as described in Section 2.6), so that it can respond to step changes in the reference values with the least overshoot and in the fastest time. Several tests are carried out in order to compare the efficiency of the proposed fuzzy controller developed in Section 5.3.1 with that of the PI_V controller. Naturally, the performance of the latter degrades by a change in the operating conditions. Various disturbances such as switching on/off a transmission line or a shunt load, or a more severe disturbance, such as a three phase short circuit, can change the operating conditions of the power system.

5.7.1. Case Study 5: Short Circuit at the PCC

A 100 *ms* three phase short circuit is applied after 1.3 sec at bus 5 in Figure 5.6. Figures 5.23-5.25 compare the performances of the two STATCOM controllers. It can be seen that the fuzzy controller damps out the oscillations faster and with less overshoot.

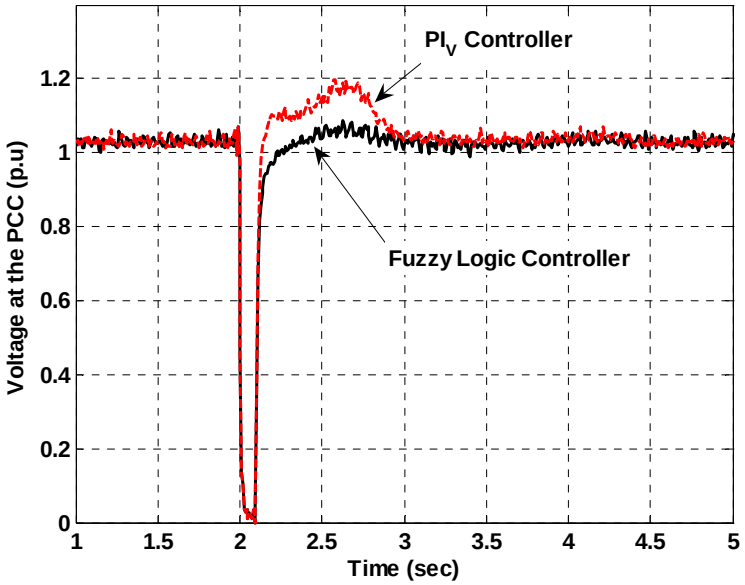

Figure 5.23. Bus 5 voltage (Figure 5.6) during case study 5.

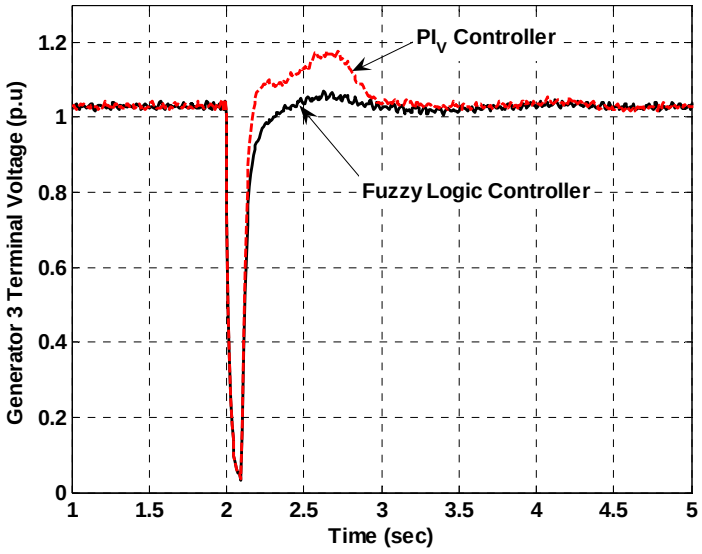

Figure 5.24. Generator 3 terminal voltage during case study 5.

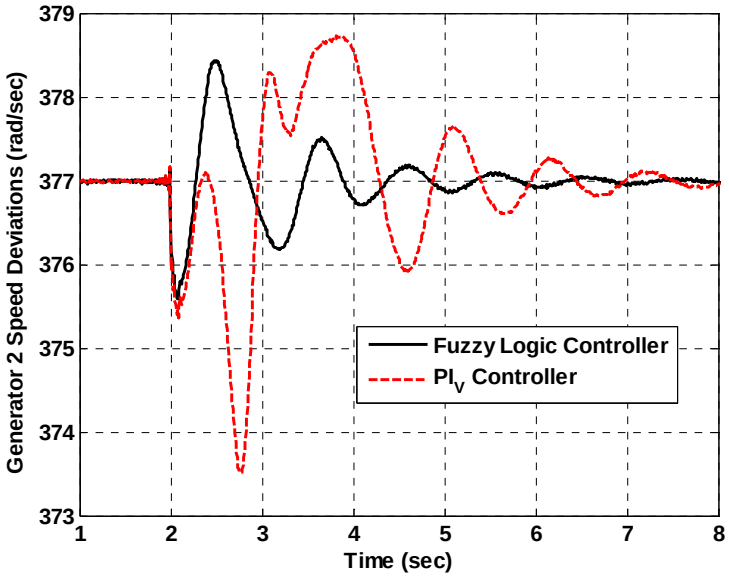

Figure 5.25. Generator 2 speed deviations during case study 5.

The performance of the two controllers can also be compared in terms of the control effort provided by each one. The reactive power injected during the fault by the STATCOM equipped with each controller in turn, is compared in Figure 5.26. These results show that the PI controller injects a considerably larger amount of reactive power into the power system, which in turn means higher currents through the inverter switches. Therefore in the case of the conventional controller, switches with higher current ratings are required.

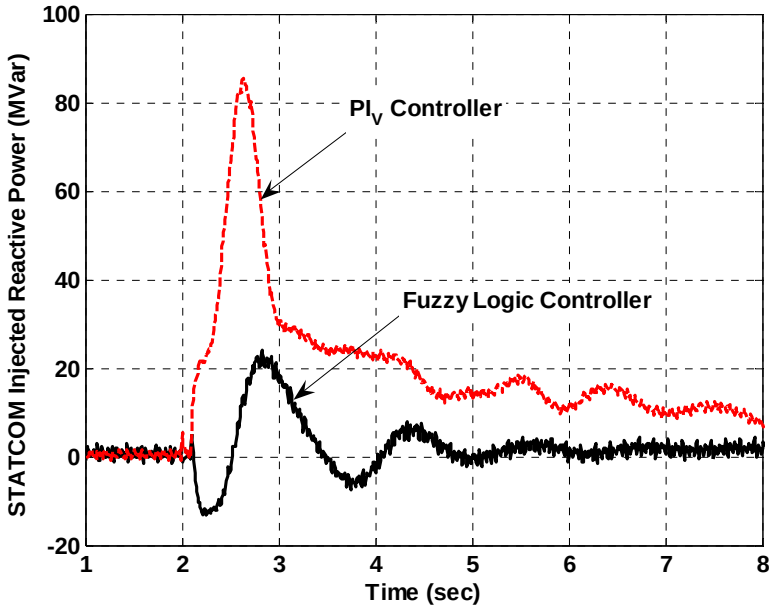

Figure 5.26. Reactive power injected by the STATCOM during case study 5.

5.7.2. Case Study 6: Short Circuit at the Load Area

The system is now subjected to a 100 *ms* three phase short circuit at the load area (bus 8 in Figure 5.6). Figures 5.27-5.29 compare the performances of the two controllers. The fuzzy controller once again provides more damping during the short circuit compared to the conventional PI_V controller. This happens since the PI_V controller is fine tuned

assuming a linear model for the system, while in a large scale disturbance such as a three phase short circuit, the system momentarily moves away from its steady state operating point and the linear control assumptions do not provide optimal performance anymore.

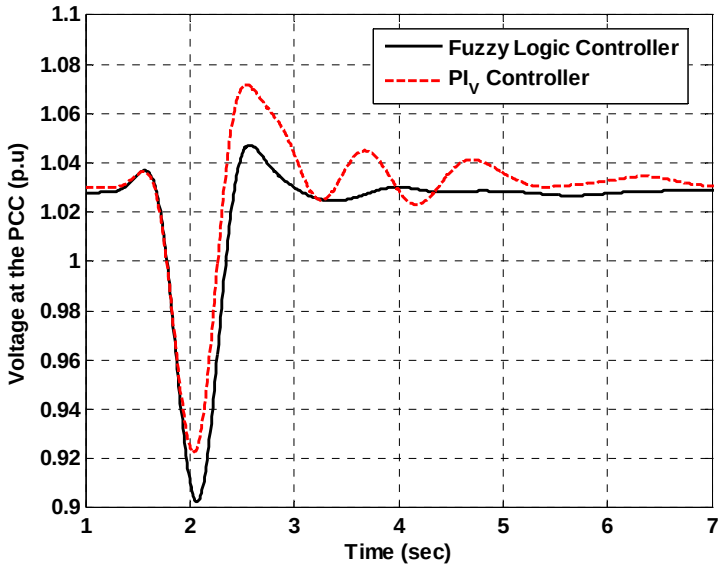

Figure 5.27. Bus 5 voltage (Figure 5.6) during case study 6.

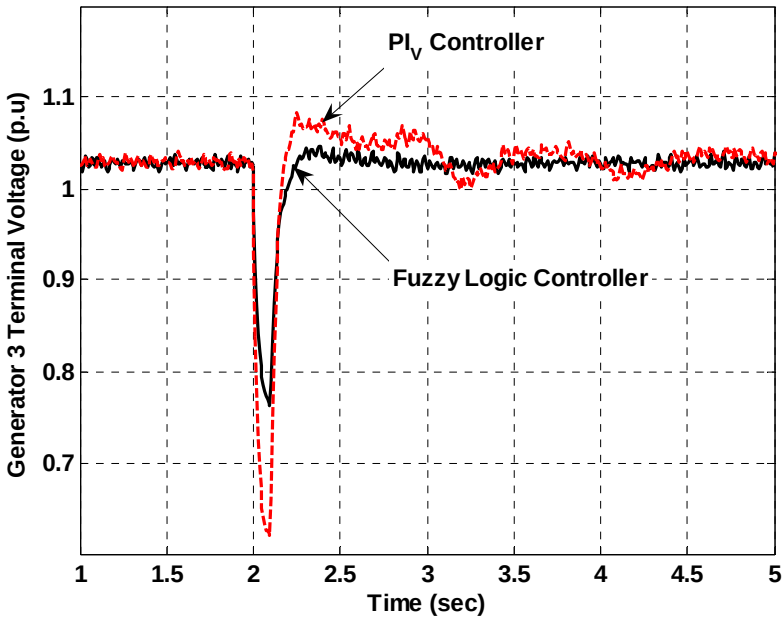

Figure 5.28. Generator 3 terminal voltage during case study 6.

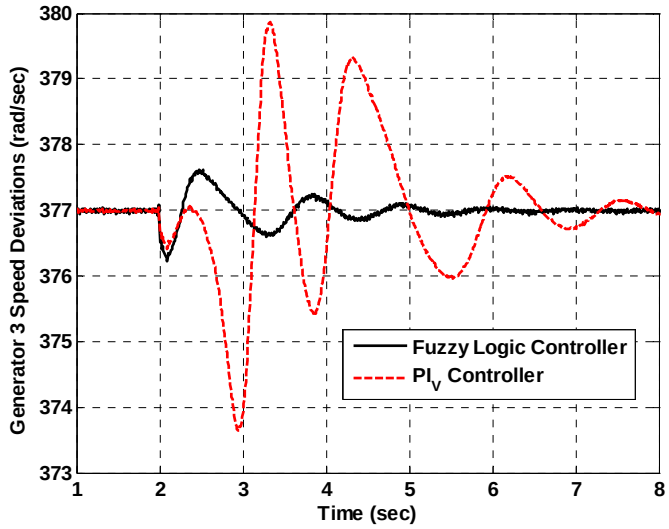

Figure 5.29. Generator 3 speed deviations during case study 6.

Figure 5.30 shows that the conventional controller responds with a much larger change in the inverter's modulation index. For cases where the inverter is working close to a modulation index of unity, such a large change might lead to over-modulation which in turn results in harmonic distortion. This has been shown in (Mohagheghi *et al.*, 2004a).

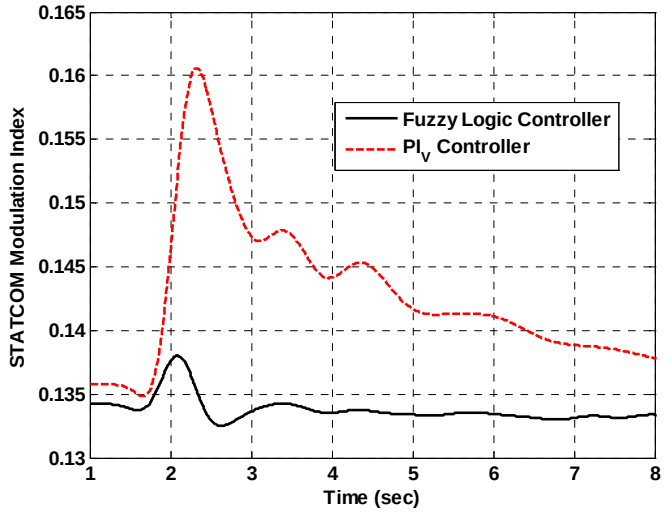

Figure 5.30. STATCOM inverter modulation index during case study 6.

Figure 5.31 shows the reactive power injected by the STATCOM, comparing the performances of the two controllers. As expected the PI_V controlled inverter injects a much larger amount of reactive power.

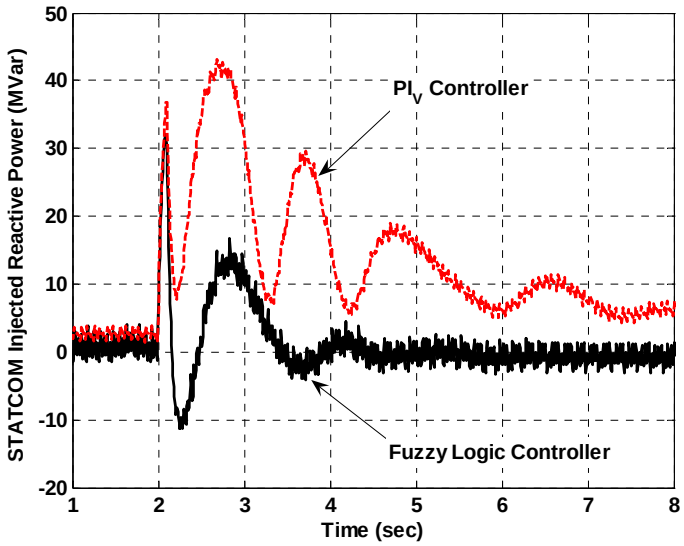

Figure 5.31. Reactive power injected by the STATCOM during case study 6.

Observing the peak compensated reactive power in cases 5 and 6, about 3 times less current rating is required when fuzzy logic control is used in comparison to the PI control.

5.7.3. Performance Index Analysis

The performance of the fuzzy controller depends on the chosen sampling frequency. In order to study the effect of the frequency on the control results, a performance index (P.I.) is defined as:

$$P.I. = \left[\frac{\sum_{k=1}^{N}(\omega_k - \omega_{Base})^2 + \sum_{k=1}^{N}(V - V_{Base})^2}{N}\right]^{-1}. \quad (5\text{-}13)$$

where N is the number of samples obtained. The performance index (P.I.) is inversely proportional to the deviations in the line voltage at the PCC and the rotor speed deviations of synchronous generator 3. It is a measure of how fast and accurate the controller can restore the system to its steady state conditions. The performance index is calculated for the fuzzy controller during a 100 *ms* three phase short circuit at bus 5 in Figure 5.6. Only the voltage at bus 5 and the speed of the generator 3 are taken into account, since these are the quantities most affected by the STATCOM performance. Table 5.2 summarizes the results. As expected, by increasing the sampling frequency (on the DSP card) the performance of the controller is improved and the deviations are reduced.

Table 5.2. Performance index for different sampling frequencies.

Sampling Frequency (Hz)	Performance Index
20	0.22
50	0.19
100	0.26
200	0.28
500	0.30

5.7.4. Stability Analysis Using the Matrix Pencil Method

The stability of the closed loop system with the different controllers applied in this chapter for STATCOM control has been investigated using the *matrix pencil* method (see Appendix E). This analytical tool provides the modes of the system from the impulse response. During a short circuit, the system faces impulsive forces and the post-fault time domain response is used to find the modes of the system and thereby, the oscillation

frequencies and damping associated with each mode. This mathematical entity has been applied by many researchers in array processing and spectral estimation (Hua and Sarkar, 1990; Zoltawski, 1988).

The matrix pencil is simply a linear combination of two matrices, $Y_1 - z.Y_2$, where z is a scalar variable, and Y_1 and Y_2 are matrices that can be constructed directly from a data sequence. The matrix pencil method is computationally more efficient and more robust to noise than other methods (Venayagamoorthy *et al*., 2005b), and it predicts a matrix whose roots provide z_i (poles of the discrete system), from which the eigenvalues (continuous domain) can be determined. The number of significant modes (*M*) in the system is found from the singular value decomposition of a matrix *Y*, formed from the sampled data. Once z_i is known, the residues can be found. Details on the matrix pencil method are given in (Pereira-Filho and Sarkar, 1995; Venayagamoorthy *et al.*, 2005b; Hua and Sarkar, 1990; Zoltawski, 1988).

Different fault responses provided by the two controllers, the fuzzy controller developed in Section 5.3.1 and the PI_V controller, are investigated with the matrix pencil method and the results appear in Tables 5.3 and 5.4 that show the improvement in damping with the fuzzy logic controller with respect to the PI controller for a 100 *ms* short circuit. The number of the dominant modes of oscillation are reduced in the case of using a fuzzy controller compared to the case of a PI controlled STATCOM. The improvement is specifically considerable for the rotor speed deviations of generator 3. In Tables 5.3 and 5.4 the blank cells under frequency and damping represent the modes which are not visible under that control scheme; this means that this mode has been mitigated.

Table 5.3. Frequencies and damping of dominant modes of generator 2 for a 100 *ms* short circuit under different control schemes.

PI_V		FLC	
Freq (Hz)	Damping (%)	Freq (Hz)	Damping (%)
0.42	26.05	-	-
0.74	8.98	-	-
-	-	0.80	12.30
1.02	5.85	-	-
-	-	1.17	11.27
1.62	6.52	-	-

Table 5.4. Frequencies and damping of dominant modes of generator 3 for a 100 *ms* short circuit under different control schemes.

PI_V		FLC	
Freq (Hz)	Damping (%)	Freq (Hz)	Damping (%)
0.36	20.38	-	-
0.65	7.25	-	-
-	-	0.89	7.23
0.99	15.90	-	-
1.29	13.71	-	-
2.00	4.32	-	-

5.8. Conclusions

In this chapter, a proportional-integrator Takagi-Sugeno fuzzy controller has been designed for a static compensator connected to a multimachine power system. The performance of the proposed fuzzy controller has been further improved by applying an indirect adaptive scheme for updating its output parameters. A RBF based neuroidentifier has been trained in order to model and track the ever changing power system dynamics

instead of having a fixed mathematical model of the plant. The neuroidentifier has been trained online and does not require large amounts of offline data.

The effectiveness of both the proposed fuzzy controllers has been compared with that of the tuned conventional PI_V controller for the STATCOM. Simulation results indicate that the fuzzy controllers are more effective in responding to small scale disturbances such as step changes to the STATCOM voltage reference, as well as to the large scale faults such as three phase short circuits.

CHAPTER 6

STATCOM ADAPTIVE CRITIC DESIGN BASED NEUROCONTROLLER

6.1. Introduction

In the previous chapters, the advantages of the intelligent control techniques over linear techniques and traditional nonlinear adaptive controllers were discussed. Fuzzy logic based controllers have, for example, been used for controlling a STATCOM (Mohagheghi *et al.*, 2004a; Morris *et al.*, 2003). The performance of such controllers can further be improved by adaptively updating their parameters. In the previous chapter an indirect adaptive fuzzy controller was developed and tested for a STATCOM in a multimachine power system. Mohagheghi *et al.* (2005b) also applied the controller output error method (COEM) introduced by Anderson *et al.* (1997) in order to implement an adaptive fuzzy controller for the STATCOM. Artificial neural network based indirect adaptive controllers have also been used to provide adaptive control for a STATCOM (Mohagheghi *et al.*, 2003b). However, even these indirect adaptive controllers suffer from the disadvantage of being "short-sighted". The error at one step ahead is used for updating the parameters of the adaptive controller, without considering the fact that in a real power system, the actions which take the system as close to the set-point as possible at time $(t+1)$, may end up taking the system further away from the set-point a few moments later. The basic problem is caused by a controller that is not addressing the problem of how to stay close to the desired trajectory for more than one time period into the future (Werbos, 1999), resulting in solutions that are by no means optimal or suboptimal.

The powerful and well established theory of optimal control and dynamic programming can be used as an alternative. While mathematically proven to provide an optimal control policy, this technique has its own disadvantages. Solving the dynamic programming algorithm in most of the cases is not feasible. Even a numerical solution requires overwhelming computational effort, which increases exponentially as the size of the problem increases (curse of dimensionality). These restrictive conditions lead the solution to a suboptimal control scheme with limited look-ahead policies (Bertsekas, 2000). The complexity level is even further exacerbated when moving from finite horizon to infinite horizon problems, while also considering the stochastic effects, model imperfections and the presence of the external disturbances.

Adaptive critic designs (ACD) based neurocontrollers can overcome the above mentioned problems. These are powerful techniques designed to perform approximate dynamic programming (ADP) in the presence of noise and uncertainties, even in non-stationary cases and provide optimal control over the infinite horizon of the problem (Werbos, 1999). Such controllers do not need prior information of the plant to be controlled and can be trained online without any large amount of offline data.

In this chapter an ACD based neurocontroller is proposed for a STATCOM in a multimachine power system that provides nonlinear optimal control over the infinite horizon of the problem. The proposed controller uses the action dependent heuristic dynamic programming (ADHDP) method which is a member of the ACD family. Two case studies are presented in this paper: A 9-bus and a 45-bus power system, both with a STATCOM. The former system is depicted in Figure 4.1, while the latter power system represents a section of the Brazilian power network (Figure 6.1), which is a 45-bus, 10-

generator system, with two voltage levels of 525 kV and 230 kV respectively, 14 transmission lines at 525 kV and 41 lines at 230 kV, 24 load buses and 7 buses with shunt compensation. The total installed capacity of the 45-bus system is 8,940 MVA. The dynamics of all the generators' AVRs, exciters, turbines and governors are modeled in detail in the PSCAD/EMTDC® environment. The power system parameters and modeling details appear in Appendix B.

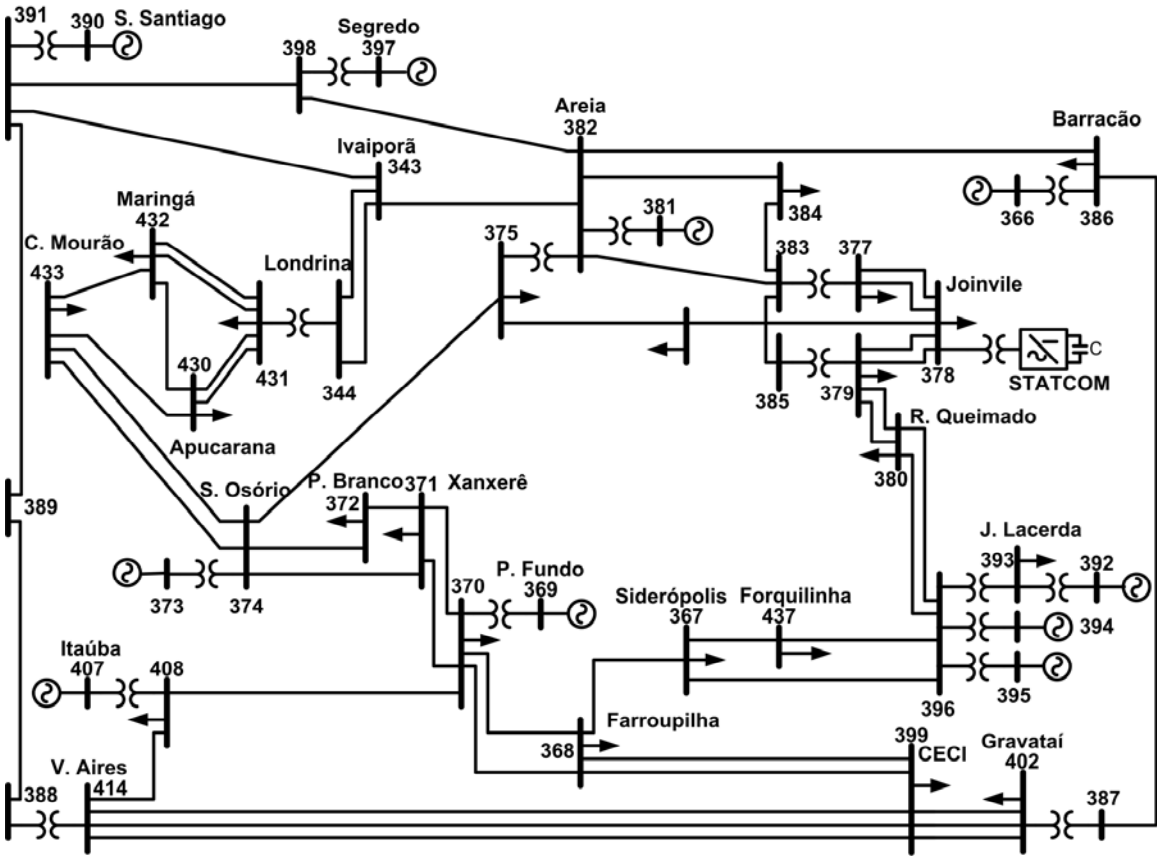

Figure 6.1. One line diagram of the 45-bus 10-generator section of the Brazilian power network.

6.2. Adaptive Critic Designs

Adaptive Critic Designs (ACD) were first introduced by Werbos in (1974), and later in (1994) and by Widrow in the early 70's (Widrow *et al.*, 1973). Werbos later on proposed a family of approximate dynamic programming (ADP) designs (Werbos, 1990). These are neural network based techniques capable of optimizing a measure of utility or goal satisfaction, over multiple time periods into the future, in a nonlinear environment under conditions of noise and uncertainty; in other words they perform maximization/minimization of a predefined utility function over time (Werbos, 2000; Prokhorov and Wunsch, 1997).

A utility function $U(t)$ along with an appropriate choice of a discount factor should be defined for the ACD neurocontroller. At each time step t, the plant output (a vector of measured variables) $X(t)$ are fed into the controller, which in turn generates a policy (control signal $A(t)$) in a way that it optimizes the expected value function over the horizon time of the problem which is known as the *cost-to-go function J* given by Bellman's equation of dynamic programming (Bertsekas, 2000) as:

$$J(t) = \sum_{k=0}^{\infty} \gamma^k U(t+k), \tag{6-1}$$

where $U(.)$ is the utility function and γ is a discount factor for finite horizon problems ($0 < \gamma < 1$). A discount factor of zero uses the present value of the utility function as the optimization objective (similar to the minimization of one step ahead error, such as the indirect adaptive control scheme), while a discount factor of unity considers all the future values of the utility function equally important and is more suitable for the infinite horizon problems.

Essentially, ACD based controllers are based on three different mathematical theories: adaptive control, optimal control and reinforcement learning. Two major categories of the ACD family include the *model based* ACD design, where a model of the plant to be controlled is required in order to train the neurocontroller, and the action dependent ACD designs (ADACD), which is a *model free* approach. In an ADHDP based ACD neurocontroller two different neural networks are used:

- *Critic network*; a neural network trained to accomplish the task of dynamic programming by approximating the true cost-to-go function *J* with no prior knowledge of the system. Moreover it avoids the curse of dimensionality that occurs in some cases of classical dynamic programming based optimal control (Werbos, 2000).
- *Action network*; which functions as a controller and is trained to send the optimum control signals to the plant, resulting in minimization/maximization of the function *J* over the time horizon of the problem.

The ADHDP-based ACD neurocontroller configuration with the Critic and Action neural networks is shown in Figure 6.2, where $X(t)$ is the vector of the plant outputs (i.e., the line voltage deviations), X_{ref} is the vector of the plant reference signals (i.e., the STATCOM line voltage reference), and $A(t)$ is the vector of the controller outputs (i.e., the inverter modulation index m_a). Both the neural networks are three layer feedforward multilayer perceptron (MLP) type neural networks having a single hidden layer with hyperbolic tangent activation functions; and the backpropagation algorithm is used for training these networks and updating their synaptic weight matrices (Haykin, 1998).

The simulation step size of 100 μs is selected for the PSCAD simulations, while the sampling time for both the neural networks is 2 *ms*.

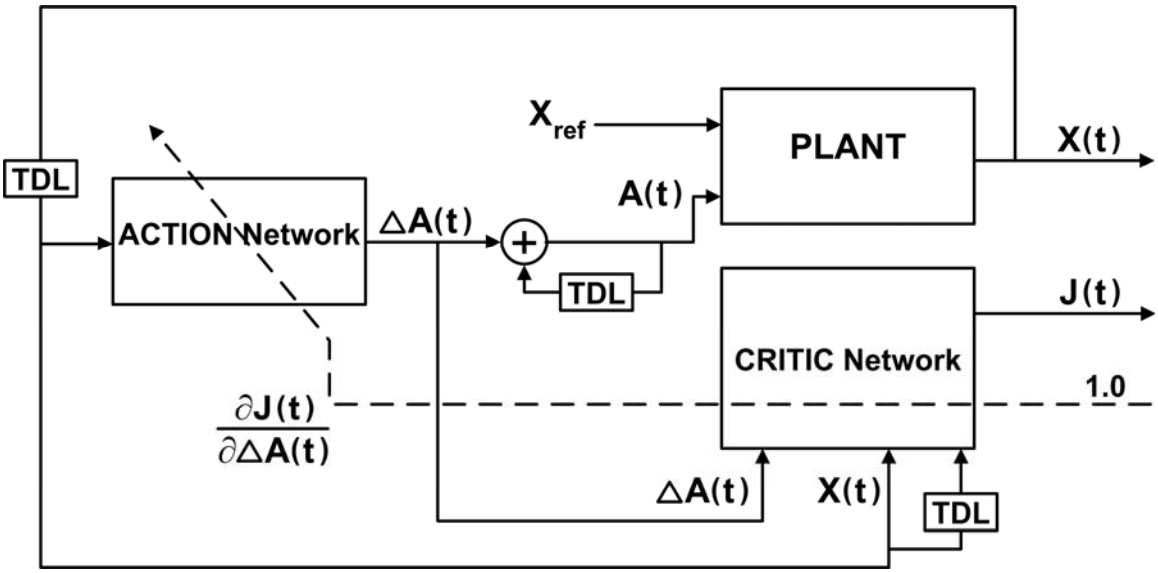

Figure 6.2. Schematic diagram of a ADHDP based ACD neurocontroller.

6.3. STATCOM Neurocontroller Structure and Training

6.3.1. Critic Network

The Critic network learns to approximate a cost-to-go function *J* using the plant input and outputs which are fed to the Critic from the plant and the Action network. If, for a controllable plant, this neural network converges to the correct cost-to-go function, the controller will stabilize that plant; in other words provided the Critic network converges correctly, the neurocontroller will provide universal stable adaptive control (Werbos, 2000).

The ADHDP Critic network structure is shown in Figure 6.3. It is a three layer MLP neural network that predicts the value of the corresponding cost-to-go function *J* at time *t*,

given the plant output ΔV (see Figure 4.1) at times *t* and its two delayed values at (*t*-1) and (*t*-2), along with the Action network output at time *t* as the input vector. The number of neurons in the hidden layer of the Critic network is heuristically chosen to be seven.

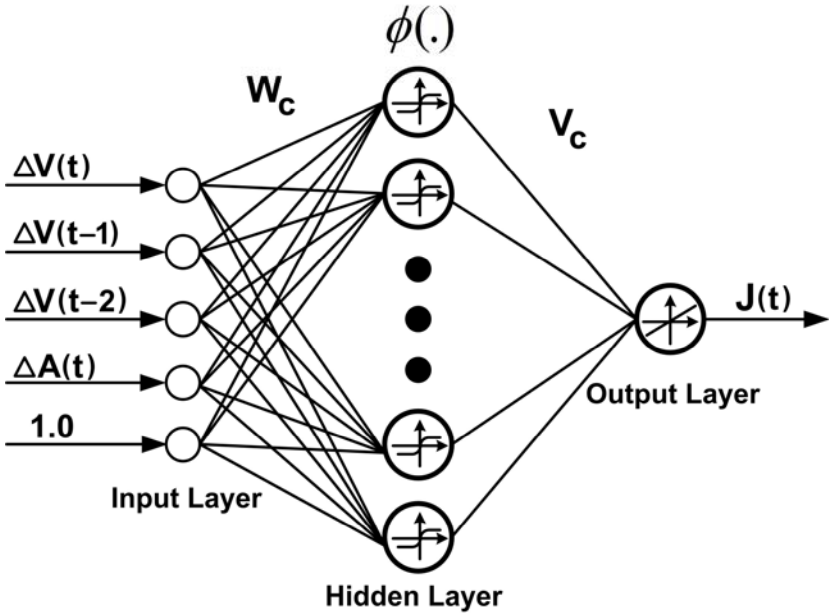

Figure 6.3. Schematic diagram of the ADHDP Critic network.

Bellman's equation in Equation (6-1) indicates that:

$$J(t-1) = U(t-1) + \gamma \times U(t) + ... = U(t-1) + \gamma \times J(t). \tag{6-2}$$

Therefore the instantaneous error can be defined as a function of two successive values of the cost-to-go function *J*. This is normally referred to as the temporal difference error:

$$e_C(t) = \gamma \times J(t) + U(t-1) - J(t-1). \tag{6-3}$$

The objective of training the Critic network is to minimize the following mean-squared error:

$$E_C(t) = \frac{1}{2} \times e_C(t)^2 . \tag{6-4}$$

A steepest descent method is used for updating the synaptic weights matrices W_C and V_C of the Critic network (Figure 6.3) in the negative direction of the derivative of the error function defined in Equation (6-4):

$$W_C(t+1) = W_C(t) - \eta_C \times \frac{\partial E_C(t)}{\partial W_C(t)}, \tag{6-5}$$

where η_C is the Critic network learning rate and the weight update equation can be rewritten as in Equation (6-6):

$$\frac{\partial E_C(t)}{\partial W_C(t)} = \frac{\partial E_C(t)}{\partial J(t)} \times \frac{\partial J(t)}{\partial W_C(t)} . \tag{6-6}$$

The same procedure can be applied for updating the weight matrix V_C. For the detailed explanation of the backpropagation training algorithm, the reader is referred to (Feldman, 1996). Figure 6.4 illustrates the schematic diagram of training the Critic network. The two Critic networks shown are identical and they undergo the same weight update. One network predicts the real time value of the cost-to-go function *J* at time *t* whereas the second one predicts its value at time $(t-1)$.

$U(t)$ is the utility function that defines the optimization objective of the optimal neurocontroller. Selection of the utility function has a major impact on the performance and the convergence of the ACD controller. Lendaris and Neidhoefer (2004) have reviewed the common approaches for selecting the utility function. A unipolar function, as the absolute value of the linear combination of the present and past values of the plant output is selected in this work, which fits the training procedure of the Critic and Action networks best. The selected utility function for this study is given in Equation (6-7):

$$U(t) = |\Delta V(t) + \Delta V(t-1) + 4.0 \times \Delta V(t-2)|. \quad (6\text{-}7)$$

Care should be taken that during the training of the Critic and Action networks, all the natural modes of the system should be excited. This is ensured by adding a pseudorandom binary signal (PRBS) disturbance to the plant input $A(t)$ in Figure 6.4. The PRBS is a randomly generated external signal which in this study is a combination of three different frequencies 0.5, 1 and 2 Hz. The magnitude of the PRBS signal is selected in a way that it provides up to $\pm 5\%$ deviations in the plant output.

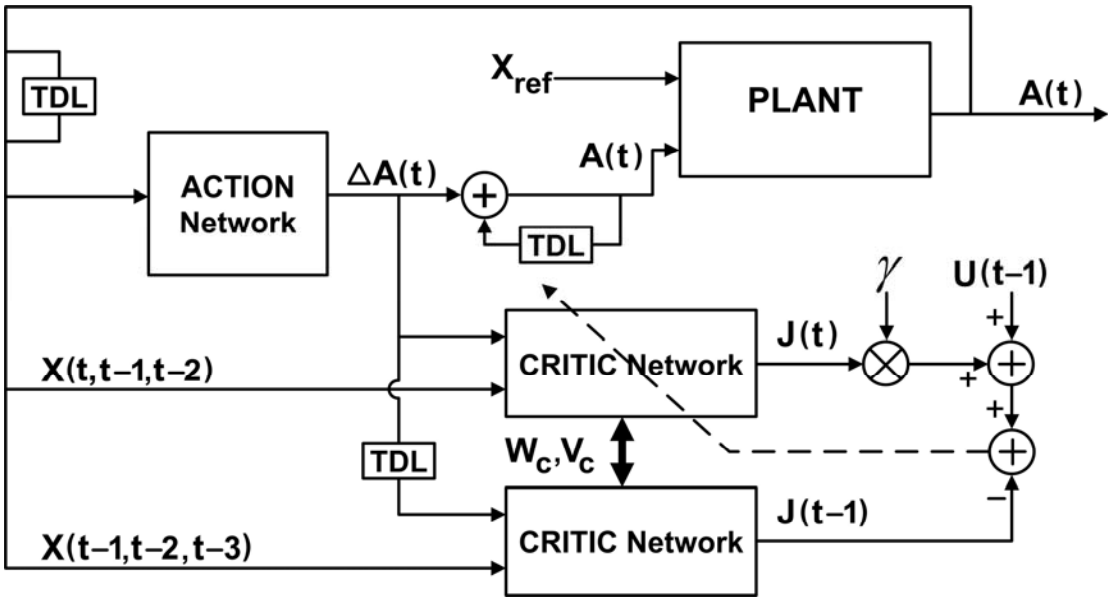

Figure 6.4. Critic network training.

The power systems in Figures 4.1 and 6.1 are run with the STATCOM PI controllers in the circuit in order to reach the steady state, at which point the PI controller outputs are constant. The STATCOM PI controller for the line voltage (PI_V) is now deactivated and its constant output m_a is applied to the plant in addition to the PRBS signal generated from an external source (switch S in Figure 4.9 is moved from position 1 to 3). More

details are explained in (Mohagheghi *et al.*, 2003a). The Critic network training error is formed as in Equation (6-3) and the weight update Equation (6-5) is applied to the Critic network for updating its synaptic weight matrices. The training process is started with a low discount factor of 0.2, and after the Critic weights have converged, the discount factor is increased to 0.5 and ultimately to 0.8. The Critic network generates output values that are used to train itself (Figures 6.2 and 6.4). As a result, at the early stages of the training process, its output may appear to be equivalent to noise, therefore this process of changing the discount factor helps the Critic network learn the dynamics of the cost-to-go function more accurately and faster (Lendaris and Neidhoefer, 2004).

A learning rate annealing process is applied for training the Critic network in which a preliminary learning rate of 0.02 is selected for the first stages of training the Critic, and this value is linearly reduced to 0.005 as the training proceeds.

6.3.2. Action Network

The Action network optimizes the overall cost over the time horizon of the problem (minimizing the function *J*) by providing an optimal control input to the plant. The network in Figure 6.5 consists of a MLP neural network with seven neurons heuristically chosen in the hidden layer. The overall input vector consists of the values of the plant output ΔV at times *t*, $(t-1)$ and $(t-2)$, and in turn it generates the control signal $\Delta A(t)$ for the plant (Figure 6.5).

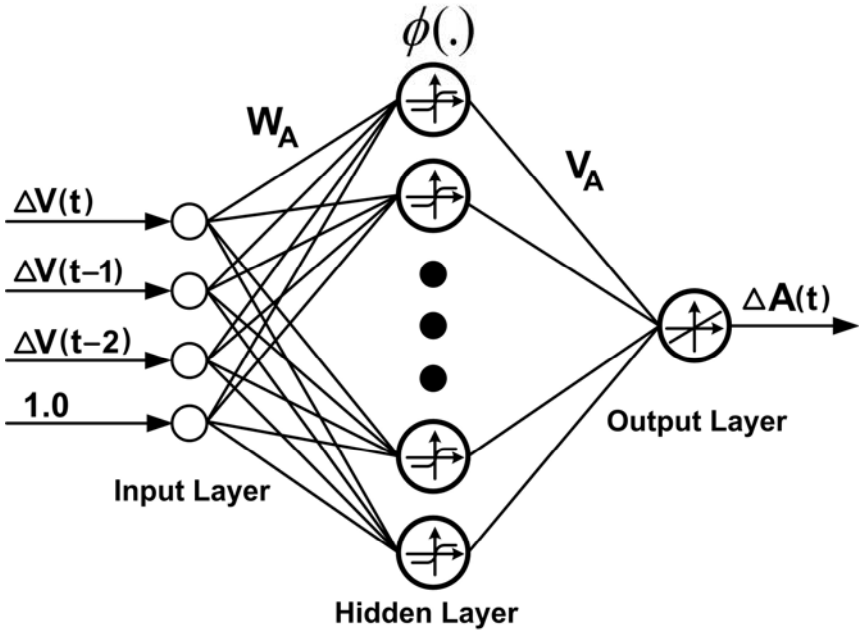

Figure 6.5. Schematic diagram of the ACD Action network.

In order for the Action network to be able to minimize the cost-to-go function over the infinite horizon of the problem, it should be trained with the following error signal:

$$e_A(t) = J^*(t) - J(t), \tag{6-8}$$

where $J^*(t)$ is the desired value for the cost-to-go function, which is zero in the case of dealing with deviation signals. The mean-squared error function $E_A(t)$ in Equation (6-9) is used as the objective function for executing the backpropagation algorithm:

$$E_A(t) = \frac{1}{2} \times e_A^2(t). \tag{6-9}$$

Therefore the Action network input weights are updated according to (6-10):

$$W_A(t+1) = W_A(t) - \eta_A \times \frac{\partial E_A(t)}{\partial W_A(t)}, \tag{6-10}$$

where η_A is the Action network learning rate. Using the chain rule for the partial derivatives of the error function, the weight update equation can be expanded to:

$$\frac{\partial E_A(t)}{\partial W_A(t)} = \frac{\partial E_A(t)}{\partial J(t)} \times \frac{\partial J(t)}{\partial \Delta A(t)} \times \frac{\partial \Delta A(t)}{\partial W_A(t)}. \tag{6-11}$$

The first term on the right hand side of Equation (6-11) is equal to $J(t)$ and the last term can be derived using the backpropagation algorithm equations (Feldman 1996), while the second term is calculated by backpropagating the constant 1.0 through the Critic network (see Appendix D). A similar approach can be used for the output weight matrix V_A. This training scheme is widely used for training ACD neurocontrollers in the literature (Govindhasamy *et al.*, 2005; Yang and Liu, 2004; Venayagamoorthy *et al.*, 2003, 2002; Si and Yang, 2001).

It is also shown in Figure 6.2 that the instantaneous output of the Action network $\Delta A(t)$ is added to the sum of the previous outputs in order to generate the final control signal $A(t)$. This ensures a "proportional-integrator" type structure for the ACD neurocontroller and allows it to deal with the actual signals and not the deviations.

With the Critic network already converged, the Action network is trained online, in other words it is controlling the plant while being trained. This raises a stability issue for the controller. For an online training scheme it is always beneficial to have *a priori* information about the plant dynamics and control scheme. With this information already used for initializing the controller, the controller can provide stable control as it enhances its performance towards optimality. However, providing this initialization information, and interpreting it for a neural network, is difficult, if not impractical. Therefore, in order to ensure that the randomly initialized controller does not move the plant towards instability, several heuristic precautions can be taken as a rule of thumb (Lendaris and Neidhoefer, 2004). The Action network in this study begins the training process, as it

tries to control the system at steady state, i.e., no disturbances. Small scale disturbances, such as step changes to the voltage reference of the STATCOM are applied to the system and the Action network now starts learning the new dynamics as a result of these step change disturbances. Once satisfactory performance is achieved by the neurocontroller, it is now ready for large scale disturbances like a three phase short circuit. As a rule of thumb, a ratio of 10:1 is considered for the Action network learning rate compared to that of the Critic network (Lendaris and Neidhoefer, 2004). The training procedure for the Critic and Action networks continues until both networks converge.

The step by step procedure for training the ACD neurocontroller can be summarized as follows:

1. Train the Critic network by applying PRBS disturbances to the power system,
2. Train the Critic network during large scale disturbances applied to the power system,
3. Repeat steps 1 and 2 at different operating conditions until the Critic network weights converge,
4. Train the Action network during small disturbances applied to the power system, such as step changes applied to the line voltage reference of the STATCOM,
5. Train the Action network during large scale disturbances,
6. Repeat steps 4 and 5 at various operating conditions until the Action network weights converge.

6.4. Simulation Results

6.4.1. Case Study 1: 9-Bus Multimachine Power System

Several tests are now carried out in order to evaluate the efficiency of the proposed neurocontroller compared to the conventional PI_V controller for a STATCOM connected to the multimachine power system shown in Figure 4.1.

6.4.1.1. Test A.1: Three-Phase Short Circuit at the Load Area

It was previously mentioned that an indirect adaptive controller might suffer from being short-sighted. In other words, the controller might generate a large control signal in order to compensate for the estimated error at time $(t+1)$, but this control effort might take the system farther away from the set-point in the future. In this section a 100 *ms* three phase short circuit is applied at the shunt load (bus 8 in Figure 4.1) and as a result of that the load and the transmission line connecting buses 7 and 8 are disconnected. Figures 6.6-6.7 compare the performance of the PI_V controller, the ACD neurocontroller and the adaptive Takagi-Sugeno fuzzy controller described in Chapter 5. The PI_V controller is equipped with an anti-windup limiter (see Figure 5.14) that clamps the modulation index of the inverter at 0.8 in order to avoid over-modulation and reduce the STATCOM size.

Figure 6.6 shows that the ACD neurocontroller is effective in damping the system oscillations, whereas the fuzzy controller is slow to damp out the oscillations and it causes a considerably larger overshoot than the PI_V. This is an example of a case where one step ahead compensation (provided by the indirect adaptive controller) results in poor performance.

The controllers can also be compared in terms of the control effort provided by each one in order to deal with the fault. Figure 6.7 shows that the neurocontroller damps out

the oscillations with the least amount of reactive power injection and therefore less current through the inverter switches.

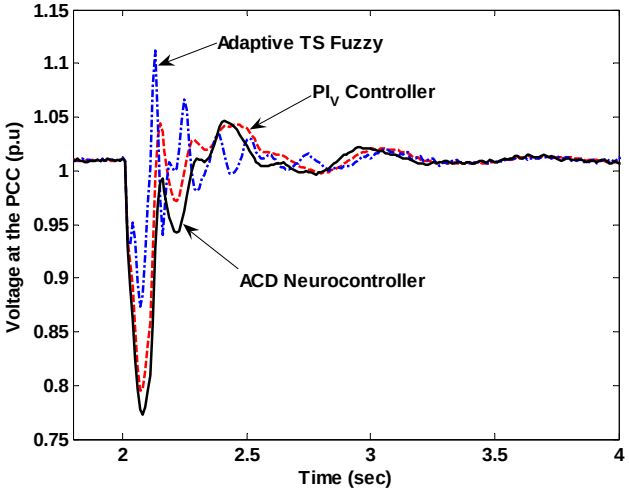

Figure 6.6. Voltage at bus 5 (Figure 4.1) during test A.1, with an anti-windup limiter for the PI_V controller.

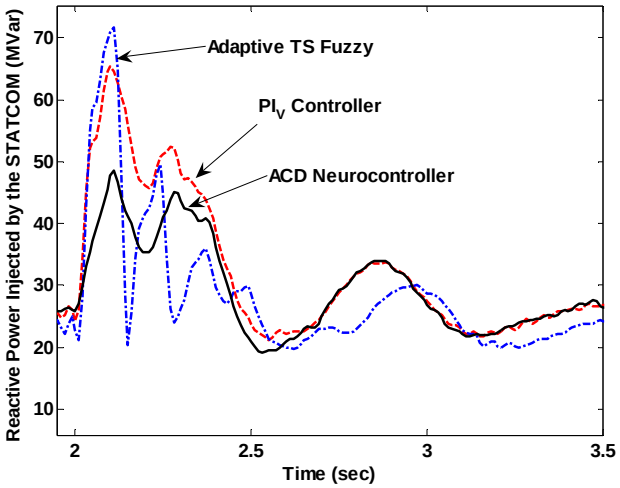

Figure 6.7. Reactive power injected by the STATCOM during test A.1, with an anti-windup limiter for the PI_V controller.

The next sections only show the responses of the ACD Neurocontroller and the PI_V controller. The adaptive fuzzy controller results for the similar tests are reported in sections 5.5.1 to 5.5.4.

6.4.1.2. Test A.2: Step Changes in the STATCOM Voltage Reference

In the second test ±5% step changes are applied to the line reference voltage of the STATCOM line voltage controller and the performances of the two controllers (PI_V and the ACD neurocontroller) are shown in Figure 6.8, which indicates that the proposed ACD neurocontroller is faster in responding to the step changes in the reference signal and it achieves this with a small overshoot. A PI controller with a larger gain (smaller time constant) can perform faster than the PI_V results shown in Figure 6.8; however, such a controller will cause large overshoots during large scale disturbances such as three phase short circuits and might even cause instability in the system.

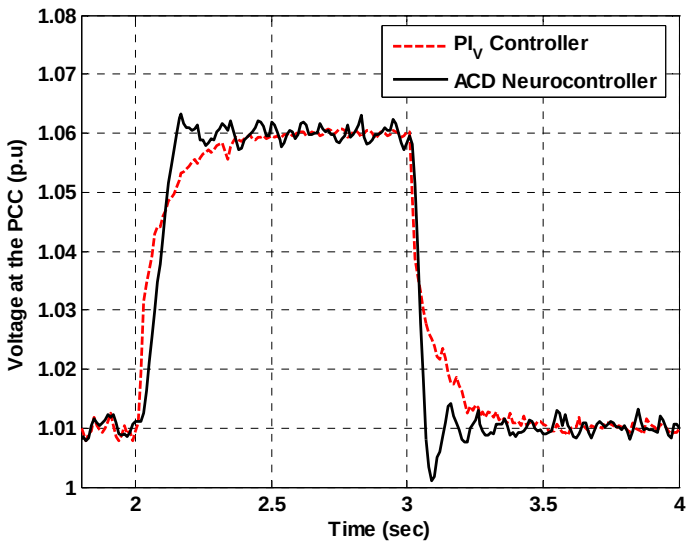

Figure 6.8. Voltage at bus 5 (Figure 4.1) during test A.2.

6.4.1.3. Test A.3: Three-Phase Short Circuit at Generator 3 Terminals

In another test a 100 *ms* three phase short circuit is applied to the terminals of the generator 3 at bus 5 (Figure 6.1). The generator is disconnected as a result of the fault and

it is reconnected to the system 50 *ms* after the fault is cleared, that is, 150 *ms* after the fault occured. The PI_V controller is equipped with the anti-windup limiter of Figure 5.14, which clamps the modulation index of the STATCOM inverter at 0.8. Figure 6.9 shows the performance of the two controllers during the transient condition. Although the PI_V controller with the anti-windup limiter is efficient in restoring the power system to steady state conditions, the ACD neurocontroller achieves this with a smaller overshoot.

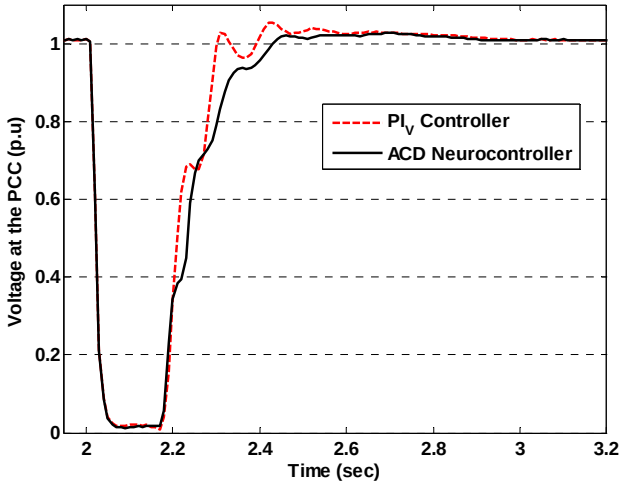

Figure 6.9. Voltage at bus 5 (Figure 4.1) during test A.3, with an anti-windup limiter for the PI_V controller set at 0.8.

Figure 6.10 compares the amount of reactive power injected by the STATCOM when it is controlled by the ACD neurocontroller with the cases where it is controlled by the PI_V controller with the anti-windup limiter. The simulation results clearly indicate that in this case, the PI_V controller with the anti-windup limiter manages to restore the power system to the steady state conditions with almost the same amount of reactive power injection as the ACD neurocontroller.

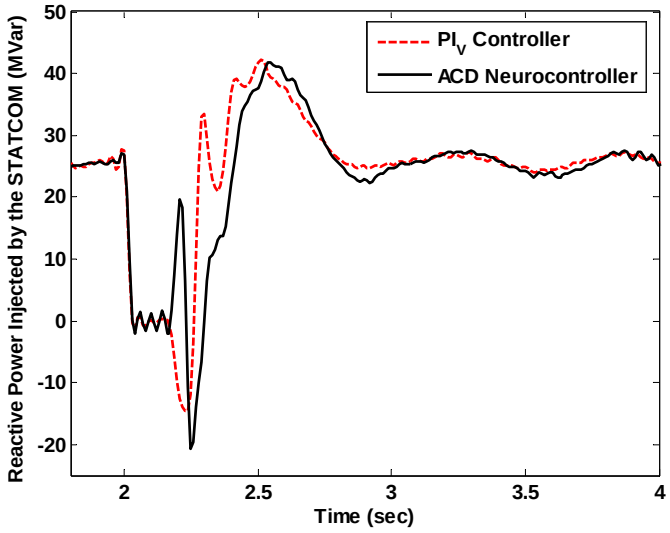

Figure 6.10. Reactive power injected by the STATCOM during test A.3, with an anti-windup limiter for the PI_V controller set at 0.8.

6.4.1.4. Test A.4: Three-Phase Short Circuit along the Transmission Line

With the same pre-fault operating conditions as the previous sections, the system is now exposed to a 100 *ms* three phase short circuit at the middle of one of the parallel transmission lines connecting buses 4 and 5 in Figure 4.1 and the line is disconnected at the moment when the fault is cleared.

Once again, the PI_V controller is considered with an anti-windup limiter that clamps the STATCOM modulation index at 0.8. Figure 6.11 illustrates the voltage at the PCC during this disturbance and it indicates that even with the anti-windup limiter, the PI_V controller imposes an overshoot on the power system, whereas the ACD neurocontroller manages to damp out the oscillation with considerably less overshoot. This is due to the fact that the fault permanently changes the configuration of the power system and therefore, the PI_V controller coefficients are not optimal anymore for the changed system. Also, Figure 6.12 shows the reactive power injected by the STATCOM for the two

controllers. The ACD neurocontroller reduces the reactive power injection of the STATCOM by almost 12 MVar and therefore, the ratings of the STATCOM switches by 30%. The cost of a typical STATCOM is roughly 100,000 $/MVA (Edris, 2006). Therefore, the ACD neurocontroller saves about 1.2 million dollars on the STATCOM's cost.

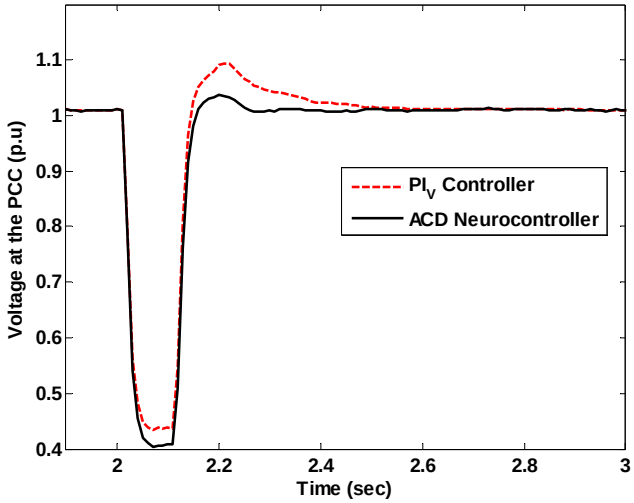

Figure 6.11. Voltage at bus 5 (Figure 4.1) during test A.4, with an anti-windup limiter for the PI_V controller set at 0.8.

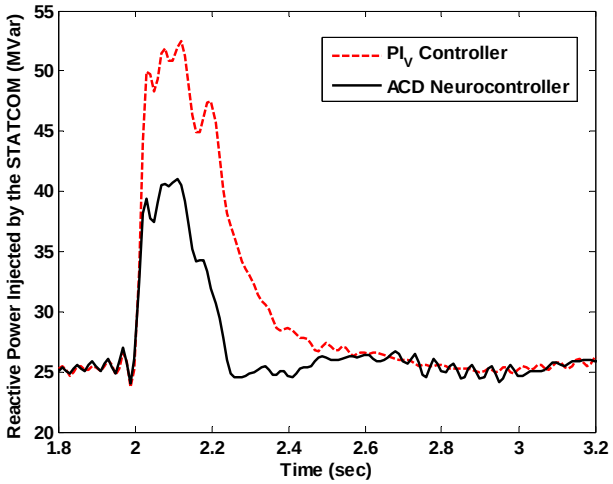

Figure 6.12. Reactive power injected by the STATCOM during test A.4, with an anti-windup limiter for the PI_V controller set at 0.8.

Figure 6.13 shows the modulation index of the STATCOM inverter for the two controllers and clearly indicates that the ACD neurocontroller exerts control effort in a smarter but smaller way than the PI_V controller with the anti-windup limiter.

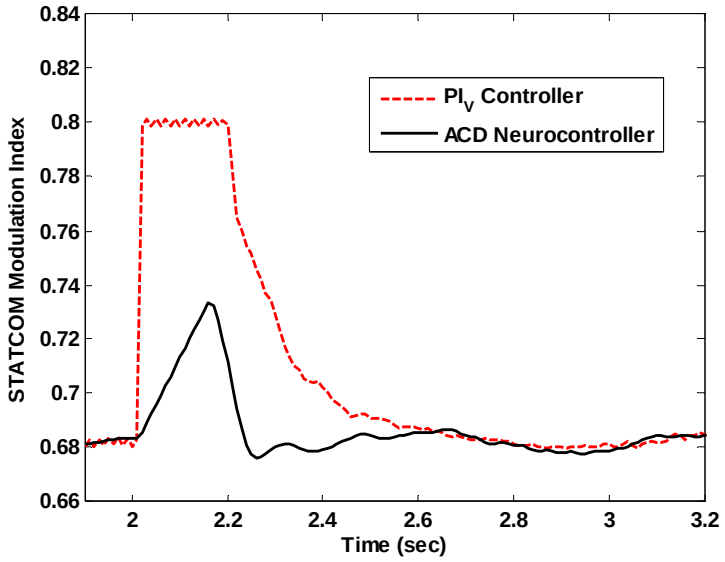

Figure 6.13. STATCOM inverter modulation index during test A.4, with an anti-windup limiter for the PI_V controller set at 0.8.

Reducing the threshold of the anti-windup limiter from 0.8 decreases the amount of peak reactive power injected by the STATCOM controlled by the PI_V controller. Figure 6.14 shows the line voltage at the PCC when the anti-windup limiter is set at 0.73, which is about the maximum reached by the neurocontroller in Figure 6.13. Also, Figure 6.15 illustrates the reactive power injected by the STATCOM. As expected the ACD neurocontroller and the PI_V with the limiter reach about the same maximum modulation index, although the neurocontroller restores the power system to steady state conditions with slightly less reactive power injection and a smaller overshoot. Reducing 1.5 MVar

of the reactive power injected by the STATCOM results in approximate saving of 150,000 dollars in STATCOM price, which is achieved at the cost of a 50% increase in the voltage overshoot and slower time response.

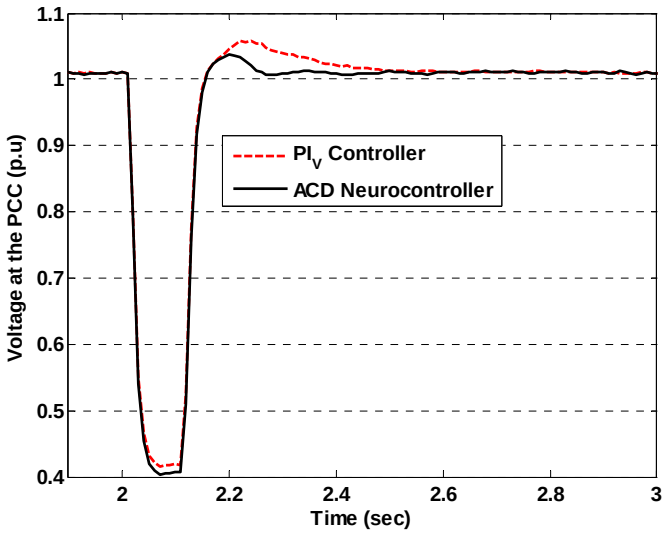

Figure 6.14. Voltage at bus 5 (Figure 4.1) during test A.4, with an anti-windup limiter for the PI_V controller set at 0.73.

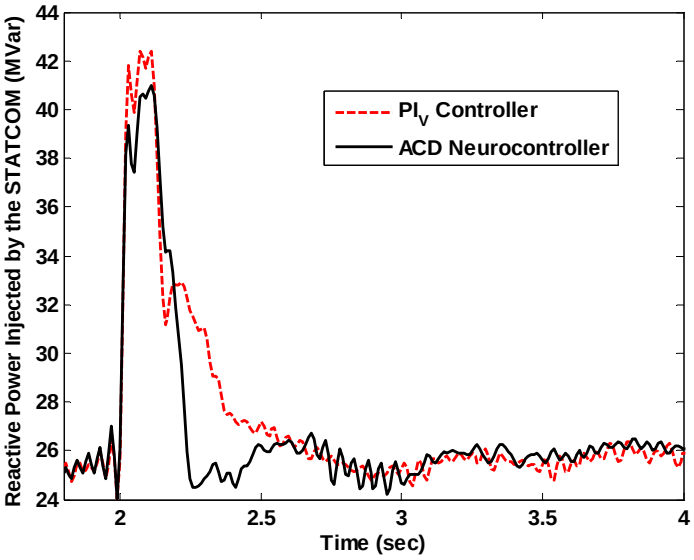

Figure 6.15. Reactive power injected by the STATCOM during test A.4, with an anti-windup limiter for the PI_V controller set at 0.73.

Reducing the threshold of the anti-windup limiter any further results in smaller reactive power injection by the PI_V controller. In this case, the result of Figure 6.14 changes to that of Figure 6.16 when the anti-windup limiter threshold is 0.7. The PI_V controller and the ACD neurocontroller initially inject the same amount of reactive power, shown in Figure 6.17, as a result of the short circuit. However, the PI_V controller fails to restore the power system to steady state condition and leads to an unstable limit cycle as shown in Figure 6.16.

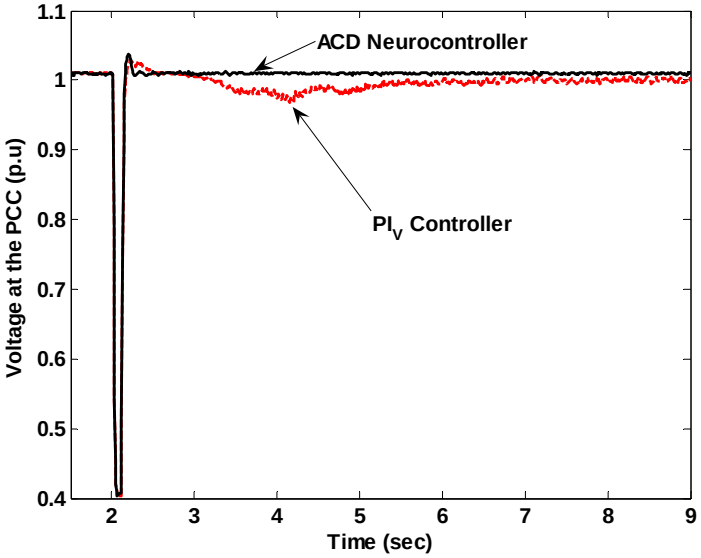

Figure 6.16. Voltage at bus 5 (Figure 4.1) during test A.4, with an anti-windup limiter for the PI_V controller set at 0.70.

Three controllers have been designed in this chapter as well as in Chapter 5 for the STATCOM in the 9-bus multimachine power system in Figure 4.1: the PI_V controller, the indirect adaptive Takagi-Sugeno fuzzy controller and the ACD neurocontroller. These controllers can be compared in terms of the amount of voltage overshoot at the PCC and the reactive power injection by the STATCOM during large scale disturbances applied to

the power system. Table 6.1 summarizes the results for the case studies A.1, A.3 and A.4[1]. In all these tests, the PI_V controller is equipped with an anti-windup limiter that is designed to clamp the inverter modulation index at 0.8. Table 6.1 clearly indicates that the ACD Neurocontroller has the best overall performance in terms of the smaller voltage overshoot and smaller reactive power injection by the STATCOM.

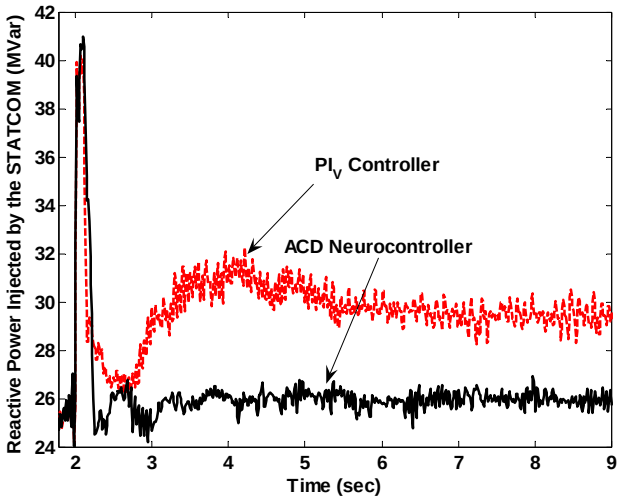

Figure 6.17. Reactive power injected by the STATCOM during test A.4, with an anti-windup limiter for the PI_V controller set at 0.70.

Table 6.1. Comparison of the three internal controllers designed for the STATCOM in the 9-bus multimachine power system, following a large disturbance.

Controller	Voltage Overshoot at the PCC (p.u)			Peak Injected Reactive Power (MVar)		
	PI_V	Adaptive Fuzzy	ACD Neurocontroller	PI_V	Adaptive Fuzzy	ACD Neurocontroller
Test A.1	1.048	1.11	1.045	65	72	48
Test A.3	1.054	1.03	1.028	42	42	41.5
Test A.4	1.12	1.06	1.04	52	52	40

[1] The results of the adaptive fuzzy controller corresponding to tests A.3 and A.4 are reported in sections 5.5.3 and 5.5.4.

6.4.2. Case Study 2: 45-Bus Multimachine Power System

Several tests are now applied to the 45 bus power system shown in Figure 6.1. No anti-windup limiter is considered for the PI_V controller in this section. This is due to the fact that in this case study the modulation index of the STATCOM during the steady state condition is relatively small (around 0.3-0.5) and does not reach saturation as a result of large scale faults and disturbances.

6.4.2.1. Test B.1: Double Phase to Ground Short Circuit

To illustrate that the STATCOM ACD neurocontroller is equally effective for unbalanced faults, a 150 *ms* double phase to ground short circuit is applied to phases *a* and *b* of bus 385. Figure 6.18 shows the three phase rms voltage at bus 378 where the STATCOM is connected to the network and that the neurocontroller is again faster in restoring the system to steady state conditions.

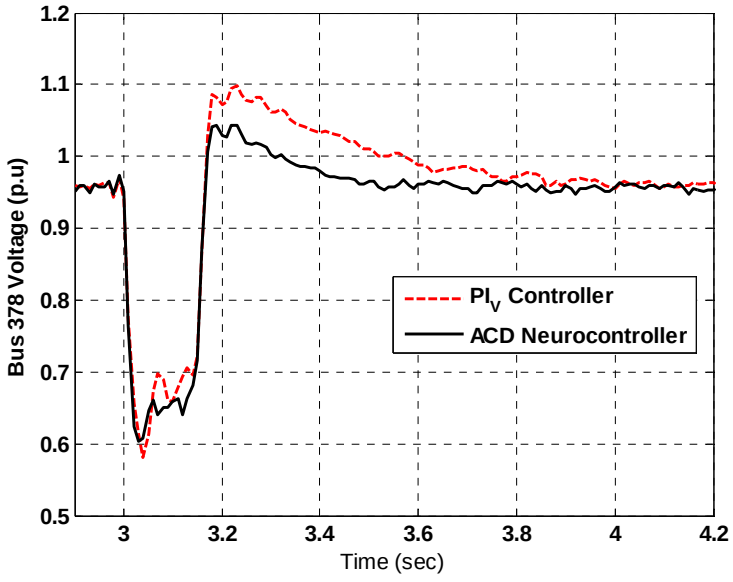

Figure 6.18. Voltage at bus 378 (Figure 6.1) during test B.1.

6.4.2.2. Test B.2: Three Phase Short Circuit along a Transmission Line

In anther test a 150 *ms* three phase short circuit occurs at the line connecting the buses 377 and 378. Figure 6.19 shows the voltage at bus 378 during the fault. Figure 6.20 shows the reactive power generated by the STATCOM in order to return the system back to stability. The neurocontroller achieves this with smaller values of power injected into the system.

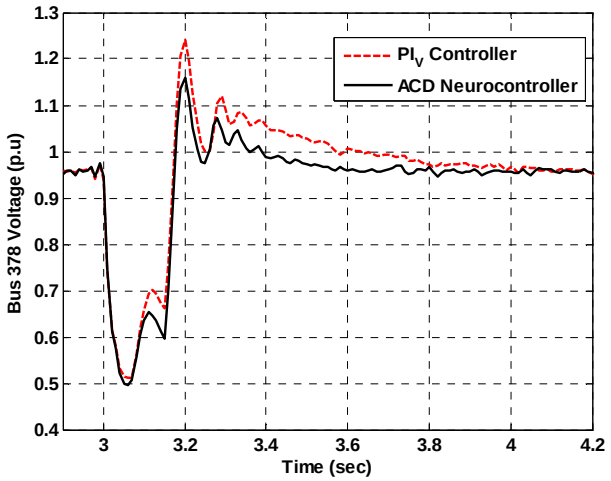

Figure 6.19. Voltage at bus 378 (Figure 6.1) during test B.2.

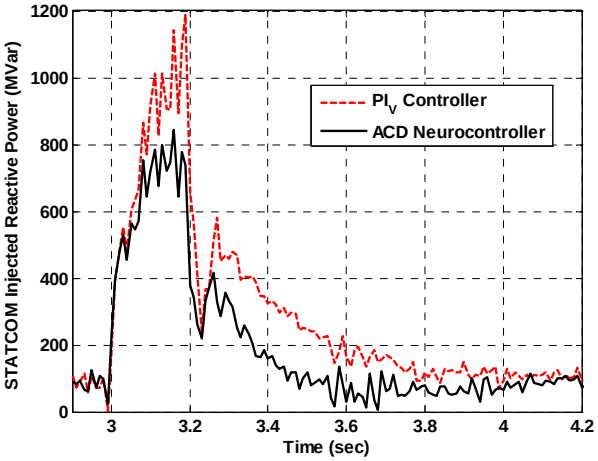

Figure 6.20. Reactive power generated by the STATCOM during test B.2.

6.4.2.3. Test B.3: Multiple Sequential Disturbances

As another test, after the above three phase short circuit is cleared, one of the lines connecting the buses 377 and 378 is disconnected and at the same time a shunt load is connected to bus 378. Figure 6.21 compares the responses of the PI_V controller and the neurocontroller to this disturbance.

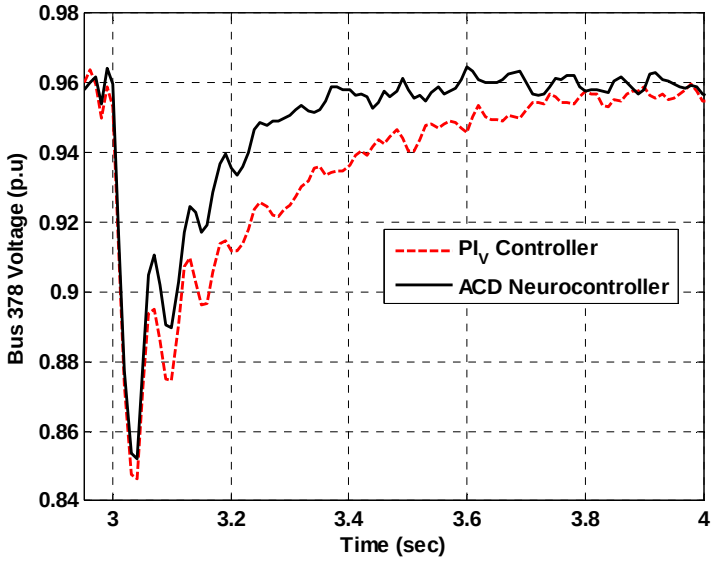

Figure 6.21. Voltage at bus 378 (Figure 6.1) during test B.3.

6.5. Conclusions

Dynamic Programming provides truly optimal solutions to nonlinear stochastic dynamic systems. However, for the majority of real life engineering problems this technique is not practical due to the curse of dimensionality. Even if practical, it will be at the cost of tremendous computational effort. Adaptive critic designs are methods that combine the concepts of Approximate Dynamic Programming with Reinforcement Learning. These are techniques capable of providing near optimal performance for the

highly nonlinear non-stationary systems in the presence of noise and uncertainty, such as a power system.

In this chapter, an action dependent HDP-based neurocontroller, a member of the ACD family, has been designed for a Static Compensator connected to a multimachine power system. Two case studies have been considered: a 9-bus 3-generator power system and a 45-bus 10-generator system as a section of the Brazilian power network and the results of several tests have been presented. The proposed ACD neurocontroller is capable of controlling the two highly nonlinear and non-stationary power systems in an optimal fashion. MLP neural networks have been used for implementing the neurocontroller, which is trained online and does not require large amounts of offline data.

The effectiveness of the neurocontroller has been compared with that of the tuned conventional PI_V controller for the STATCOM. The many simulation results all indicate that the neurocontroller is more effective in responding to small scale disturbances such as step changes to the STATCOM voltage reference, as well as to the large scale balanced and unbalanced faults including three phase short circuits and load changes.

CHAPTER 7

NEURAL NETWORK BASED WIDE AREA STATE PREDICTOR

7.1. Introduction

The previous chapters all focused on the local control of a STATCOM in a power network using computational intelligence techniques. It is assumed in all these control schemes that the information required by the controller is available in real-time. This is a valid assumption since in most of the cases, the distance between the controller and the controlled component can be short enough to be negligible in terms of data communication system. However, sometimes a centralized or hierarchical controller should be designed for a part of the power system that tries to analyze the system from a supervisory level and send the control signals to local controllers or agents. One example can be external controllers for FACTS devices, where additional measurements from the neighborhood of the device are sent to its controller so that it can perform some peripheral control objectives other than the original objective for which it was intended. Another example can be a wide area control scheme in which various measurements throughout the power grid are used to generate auxiliary control signals for the different controllers in the network.

The main challenge in the above examples and other similar problems is to create access to the real-time system data for the supervisory level controller. Even the smallest multimachine power system covers a rather wide geographical area. Typically, an appropriate means for transmitting data such as a supervisory control and data acquisition (SCADA) system is required in order to send the information from different parts of the

network to the supervisory level controller. However, even with the best communication channels there can be transport lags in sending/receiving the data that are not negligible. The more entities using a communication channel, the slower the flow of information will be. A supervisory level controller is highly dependent on the accuracy and timing of the data received, which emphasizes the need for a reliable wide area state predictor capable of estimating the states of the system in real-time.

Any system/technique designed for such a purpose should address the following issues:

- There is a transport lag associated with the communication channels used in power networks. The proposed methodology should be able to retrieve the real-time information from the received delayed values,
- The transport lag is not necessarily static. In the worst case, it can be a missing sensor or a failed communication channel. The methodology should be robust to the partial loss of information and should be able to restore the required data using the available information,
- Any changes to the network configuration can largely deteriorate the effectiveness of the state estimator. Care should be taken that the estimator adaptively adjusts itself to the ever changing nature of the power network.

In this chapter a neural network based identifier is proposed as a wide area state predictor (WASP) in a multimachine power system. Given that the network information arrives with a dynamic transport lag, the proposed WASP is able to predict the power system outputs at one step ahead, in a way that the supervisory level controller always has access to the closest estimates of the power system outputs in real-time. In addition, it

is designed in such a way that it is robust to partial loss of information due to unforeseen transport lags or missing sensors. These issues are discussed in detail in the following sections.

The proposed WASP can be located anywhere in the power system, assuming a SCADA or some other communication system is available to transmit different information/data throughout the network. There can even be more than one WASP installed at different locations in the network, so that different entities can have access to the required information simultaneously. This scheme can be especially useful in a deregulated environment, where coordinated control is required.

The power network considered in this study is the IEEE 12-bus benchmark system which is specifically designed for implementation of FACTS devices and studying their impacts on the power grid.

7.2. State Estimation: Problem Formulation and Limitations

State estimation is the traditional approach for building a real-time model of a power network. Essentially, such real-time model is a semi-static mathematical model of the dynamics of an interconnected power network. The model is derived using snapshots of system measurements at various time intervals. These data include (Monticelli, 2000):

- Analog data: magnitude and phase angle of the network buses, the active/reactive power flow through the transmission lines and injections at different buses. This part is derived from measurements throughout the network made at certain time intervals. A network topology processor (NTP) analyzes the raw analog measurements to verify the operating limits, the zero flows in the open switches

and the zero voltage across the closed switches, the connectivity of the bus sections and suchlike.

- Logical data: the status of the switching devices which is updated on a regular basis. The explicit modeling of the switches facilitates bad data analysis when topology errors are involved,
- Static network data: corresponds to the parameters and basic substation configurations. This part is necessary since the changes to the power network configuration affect the computations.

Most state estimation programs are formulated as over-determined systems of nonlinear equations and are solved as weighted least squares (WLS) problems (Schweppe and Handschin, 1974). Consider the nonlinear measurement model[1]:

$$z_j = h_j(X) + e_j, \tag{7-1}$$

where z_j is the j^{th} measurement, X is the n-dimensional state vector, $h_j(.): R^n \to R$ is a nonlinear scalar function relating the j^{th} measurement to the states, and e_j is the measurement error which is assumed to have zero mean and a variance of σ_j^2. Assuming m measurements and n states, the m/n ratio is defined as the measurement redundancy, which in most practical cases is in the range of 1.7-2.2 (Monticelli, 2000). If there are enough measurements well distributed throughout the network in such a way that state estimation is possible, then the network is said to be observable. With an adequate redundancy level, state estimation can eliminate the effect of bad data and allow temporary loss of measurements without severely exacerbating the estimation accuracy.

[1] The definitions and mathematical equations in this section are extracted from (Monticelli 2000).

The problem can now be formulated as an optimization problem: find the value of X which minimizes:

$$J(X) = \frac{1}{2}\sum_{j=1}^{m}\left(\frac{z_j - h_j(X)}{\sigma_j}\right)^2, \tag{7-2}$$

subject to:

$$\begin{aligned} g_i(X) &= 0 \quad i = 1, n_1 \\ f_k(X) &\le 0 \quad k = 1, n_2 \end{aligned} \tag{7-3}$$

where g_i and f_k are functions representing power flow quantities.

Numerical analysis methods can be employed in order to solve the nonlinear optimization problem in Equation (7-2). However, as the dimension of the power system increases, the complexity level of the solution rises. This can specifically be a problem when a large number of redundant measurements are used in a power network. Moreover, the functions $h_j(.)$ need to be adaptively adjusted as the network operating conditions and configurations change. Clearly, incorporating the updates into the mathematical formulation of the problem can be a tedious process. Dependence on the model of the process is the major disadvantage of this technique, which can weaken its robustness and practicality in real world problems. In addition, the state estimation method converges slowly, can become trapped in local minima or produce ill-conditioned and unreliable solutions (El-Sharkawi and Marks, 2003). Ideally, state estimation should run at the scanning rate; however, due to computational limitations most practical estimators run every few minutes or when a major change occurs (Monticelli, 2000).

A neural network based approach can eliminate many of the above problems. Neural network techniques can deal with massive amounts of raw data, model a system with no

need to mathematical formulation, and with additional measurements can be made robust to partial loss of data.

7.3. STATCOM in the IEEE 12-Bus Benchmark System

The power system considered in this study is the IEEE benchmark system proposed for modeling FACTS devices in a power transmission network (Jiang *et al.*, 2005). The system shown in Figure 7.1 consists of 12 buses (six 230 kV buses, two 345 kV buses and four 22 kV buses). It covers three geographical areas (Jiang *et al.*, 2005):

- *Area 1*: Predominantly a generation center with most of its generation coming from hydro power. It includes buses 1, 2, 7, 9 and 10,
- *Area 2*: Situated about 500 km from Area 1, it is a load center with a small amount of thermal generation available. It includes buses 3, 4, 5, 8 and 11. This area needs shunt compensation in the forms of shunt capacitors or active compensators for voltage support, specifically at buses 4 and 5,
- *Area 3*: Located between the main generation area (Area 1) and the main load center (Area 2), it has some generation available, but insufficient to meet the demand. The transmission system is mainly a 230 kV system with one 345 kV link between areas 1 and 2.

Load flow and dynamic stability studies showed that a STATCOM located at bus 4 can considerably alleviate the voltage problems at buses 4 and 5. The STATCOM is assumed to be primarily controlled by the scheme shown in Figure 2.7, where two decoupled PI controllers try to control the line voltage at the PCC and the dc link voltage inside the device respectively. Controlling the voltage at bus 4 is considered as the main objective of the STATCOM's one internal controller.

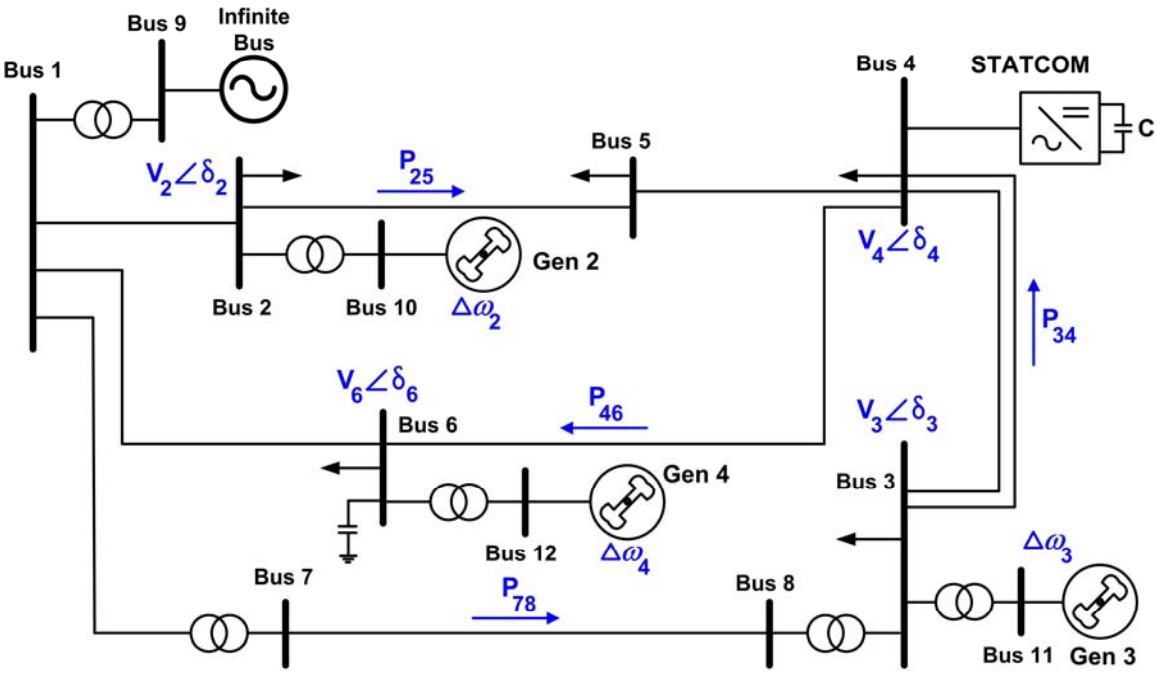

Figure 7.1. Schematic diagram of the IEEE 12-bus benchmark system with a STATCOM.

7.4. Wide Area State Predictor

The main objective of the wide area state predictor (WASP) is to receive the data from different components across the power network, process the raw information and predict the plant states in real-time. The vector of the plant inputs is considered to be:

$$U = [V_{ref}, V_{T2-ref}, V_{T3-ref}, V_{T4-ref}]^T, \tag{7-4}$$

where V_{ref} is the STATCOM line voltage reference and V_{T-ref} 's are the terminal voltage reference signals of the three synchronous generators. Clearly, the input vector U in Equation (7-4) does not include all the input signals applied to the power system. The active power reference signals to the generator turbines for example are ignored. This is due to the fact that the main objective of the supervisory level controller in this study (Chapters 8 and 9) is considered to be improving the dynamic stability of the power

system, which is normally achieved by providing auxiliary voltage references to the synchronous generators and the STATCOM. However, the ideas set forth in this chapter can also be applied to a problem with a different nature or be extended to a more general case.

The speed deviations of the three generators are considered to be the network states:

$$X = [\Delta\omega_2, \Delta\omega_3, \Delta\omega_4]^T, \tag{7-5}$$

and some other variables of the multimachine power system are considered as the vector of measurements across the power system:

$$Z = [V_2, V_3, V_4, V_6, \delta_2, \delta_3, \delta_4, \delta_6, P_{25}, P_{34}, P_{46}, P_{78}]^T. \tag{7-6}$$

where V_i and δ_i are the magnitudes and phase angles of the select buses with respect to the infinite bus in the network and P_{ij} represents the active power flow through the transmission line connecting the i^{th} and j^{th} buses. Clearly, the above mentioned choice of states and measurements is not the only possible set of states/measurements for the WASP. With a reasonable selection of variables, the WASP can be trained to learn almost any state/measurement relationship of the plant.

The input to the WASP includes the *last available values* of both the state vector and the measurement vector of the plant. The data is transmitted to the WASP through any existing data transmission means such as a SCADA system. The WASP in return predicts the values of the plant state vector in real-time.

In general, the transmitted information arrives at the control center, where the WASP is located, with a short delay. This latency is directly related to the amount of information and number of entities using the SCADA channels. For a power system covering a large geographic area, communication systems incorporating multiple relay stations will

increase the time delay, especially when the primary communication path is blocked and backup alternatives have to be used (Chow *et al.*, 2000). This transport lag can be static (fixed) or dynamic (variable). The amount of time delay due to data transmission between different devices in the network falls within different ranges when using different communication networks. The average communication delay could be as long as seconds for the internet or as short as tens of milliseconds in a fiber optic link (Taylor *et al.*, 2000). It is assumed in this study that the data will be provided for the WASP with a maximum of 100 *ms* delay.

Figure 7.2 depicts the chronology order of the tasks performed by the WASP. These tasks and the details of the various blocks in the flow chart are explained in more detail in the following chapters; nevertheless, they are briefly summarized here:

A. At every point (t) in time, the WASP evaluates the input signals. The availability of the data corresponding to $(t-\Delta t)$ is then checked, where Δt is considered to be 100 *ms*. If the data is available, it will be used in order to predict the plant outputs at time (t),

B. If for certain readings the transport lag is greater than 100 *ms* but less than 150 *ms*, the first available data will replace the corresponding delayed reading. The upper bound of 150 *ms* is heuristically chosen for replacing the unavailable data at $(t-100ms)$ with the last available data. This is due to the fact that for the IEEE 12-bus system studied in this chapter and the following ones, the period of oscillations is around 0.8-1.2 sec. Therefore, there is still a good chance that the values of the missing sensors at $(t-100ms)$ are relatively close to their corresponding values at $(t-150ms)$,

C. Using the available warning signals, the WASP determines the category of the unavailable data: *missing* or *lost*. Hereafter, the *missing* data corresponds to information in the network that is not received by the WASP, either due to failed sensors or larger than usual delays. Whereas *lost* data indicates information that is not physically available, e.g. the power flow through a transmission line that is disconnected as a result of a fault.

D. If some data are not available due to a part of power network being tripped/ disconnected, a pruning algorithm is performed to remove the information related to those specific signals from the WASP structure. The downsized WASP is now used in order to estimate the rest of the network states in real-time. More details on the pruning technique are provided in Section 7.6.1,

E. For the unavailable data categorized as missing, a separate algorithm is employed for missing sensor restoration, so that the unavailable signals are restored from the available network data. This procedure is iterated by a maximum number of times (defined by the user), and after each iteration the convergence of the computations is checked. If the computations converge the restored signals are used in lieu of the unavailable signals,

F. If the computations do not converge, the availability of those specific unavailable data is checked at $(t-250ms)$, where the delay of 250 *ms* is heuristically chosen according to the natural frequency of the system oscillations. In case of the availability of the data at this time step, those values are used instead of the missing values at $(t-100ms)$,

G. Otherwise, the missing values are considered to be zero and a warning signal is sent to the operator to indicate that the WASP is unable to predict the correct values of the unavailable data.

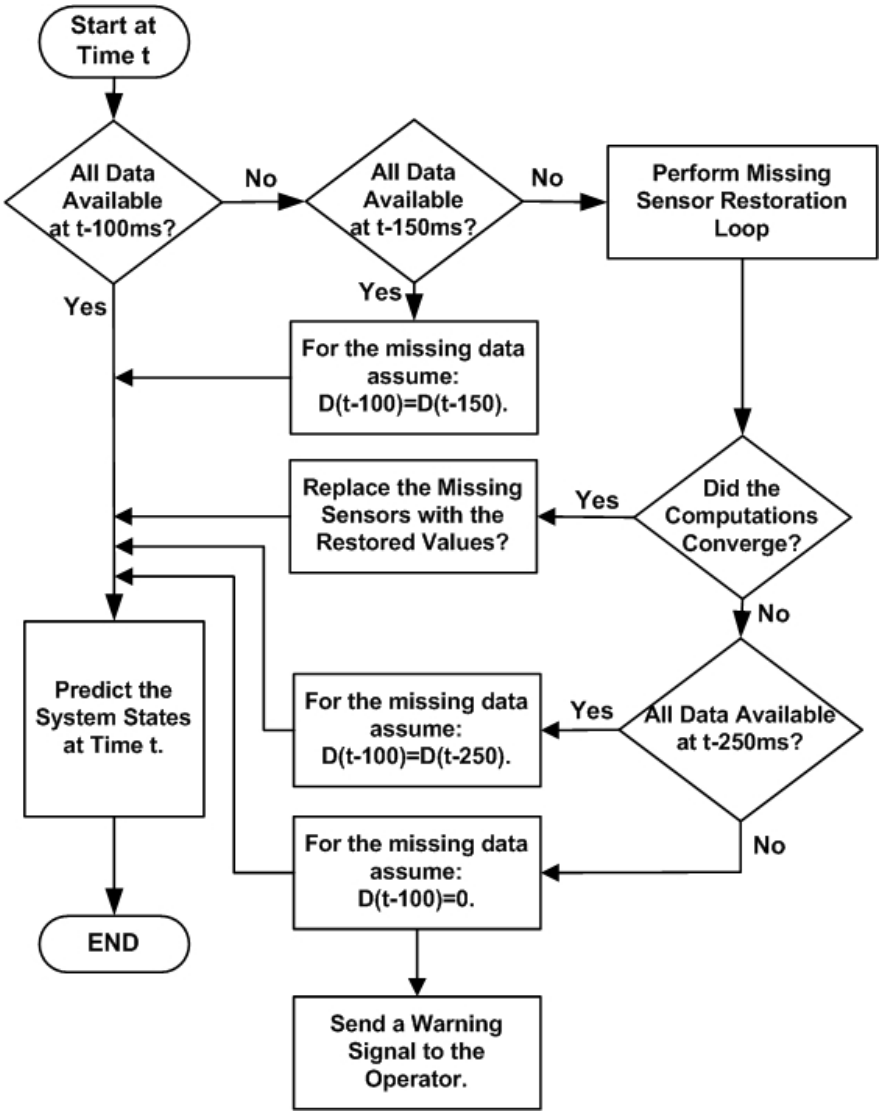

Figure 7.2. WASP flow chart.

7.5. Neural Network Based WASP

Figure 7.3 shows the schematic diagram of the proposed wide area state predictor. It consists of two major components: the neuroidentifier and the missing sensor restoration (MSR) part. The former uses the time delayed values for predicting the states of the power system in real-time, while the latter ensures that the neuroidentifier has access to the best estimates of its input vector in case of loss of information and/or failed sensors. The structure of the neuroidentifier is discussed in this chapter, while the WASP missing sensor restoration technique is explained in the next section.

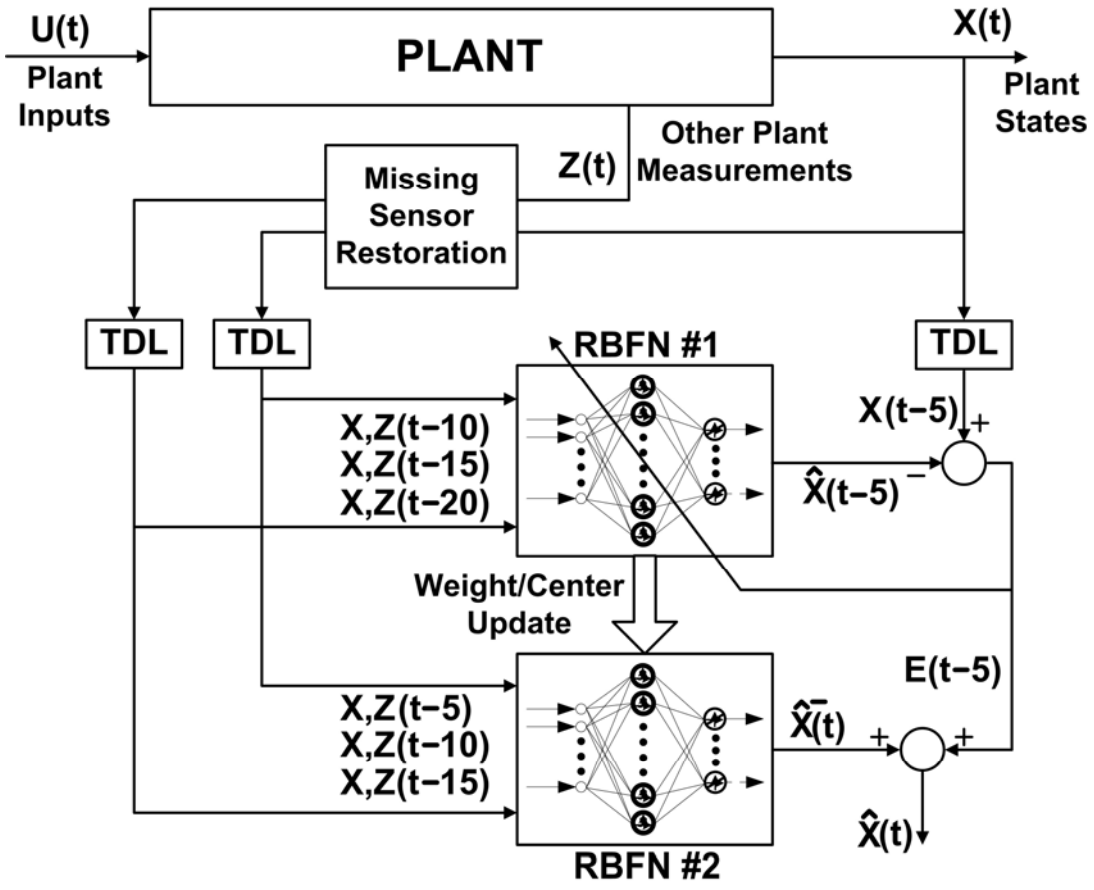

Figure 7.3. Schematic diagram of the wide area state predictor.

7.5.1. Neuroidentifier Structure

The core of the proposed WASP consists of two radial basis function (RBF) neural networks that together predict the power system states in real-time. The values of the plant state vector X and measurement vector Z at time steps $(t-\Delta t)$, $(t-2.\Delta t)$, $(t-3.\Delta t)$ and $(t-4.\Delta t)$ are used in order to predict the plant states at time step t (Figure 7.3).

With Δt considered to be 100 *ms*, a time step of 20 *ms* is selected for the RBF WASP. In other words the problem is formulated as a multiple step ahead prediction using neural networks. Figure 7.3 shows the schematic diagram for training the proposed WASP. The two RBF networks receive the same inputs but at different time steps. The first network is trained using supervised learning to track the plant states at time $(t-5)$, given the plant states and measurements at time steps $(t-10)$, $(t-15)$ and $(t-20)$. This first neural network undergoes continuous online training. The error between the output of the first neural network and the actual plant output at time $(t-5)$ is used to update its synaptic weight matrix. Backpropagation algorithm is used for training the RBF network.

At each time step the updated output weight matrix and the RBF unit centers of the first network is transferred to the second network, which receives the same type of data at time steps $(t-5)$, $(t-10)$ and $(t-15)$ and predicts the plant outputs at time step t. The predicted output of the second neural network in Figure 7.3 is also improved by adding the previous time step error to it. This is similar to the correct path of a typical Kalman filter (Welch and Bishop, 2004). It is known as the predict-correct scheme, and has been successfully reported in neural network based prediction (Govindhasamy *et al.*, 2005; Rovlak and Corlis, 1991).

With the definition of the state and measurement vectors of the plant given in Equations (7-5) and (7-6), the RBF WASP has 45 inputs plus a constant bias of 1.0 as the augmented input vector, and 3 outputs.

$$A = [X(t-\Delta t)^T, X(t-2\Delta t)^T, X(t-3\Delta t)^T, Z(t-\Delta t)^T, Z(t-2\Delta t)^T, Z(t-3\Delta t)^T]^T \qquad (7\text{-}7)$$

The number of neurons in the hidden layer is heuristically chosen to be sixty. Figure 7.4 shows the schematic diagram of the RBF neural network.

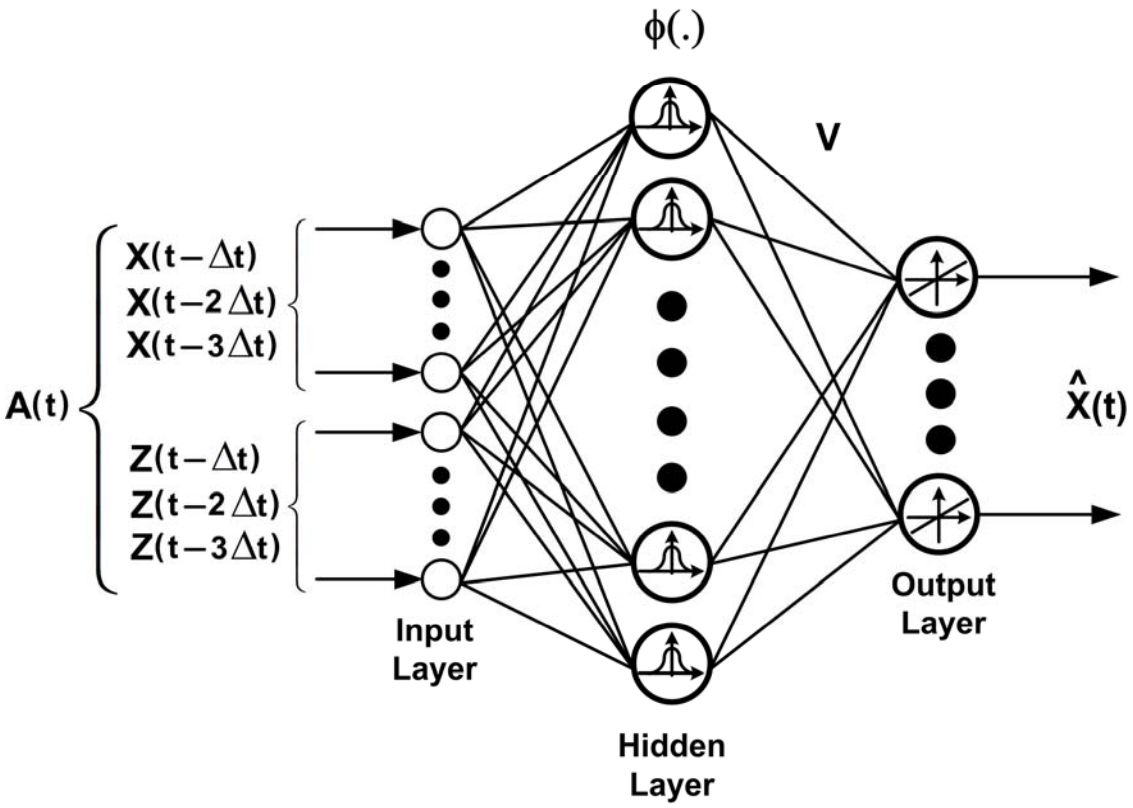

Figure 7.4. Schematic diagram of the WASP RBF neuroidentifier.

7.5.2. Training Process

Training a RBF network requires initializing and updating the centers of the RBF units and the output weight matrix (see Appendix C). The centers of the RBF units are

the heart of the neural network and are the main memory blocks of the network. Each center tries to locally model/identify a part of the input space. Several approaches are commonly used in the literature for initializing and updating these centers, where some methods update the centers online while others select an offline method to find the best location of the centers and keep them fixed during the online operation of the neural network (Haykin, 1998; Uykan *et al.*, 2000).

In the specific problem of power system state prediction, both these methods have disadvantages:

- The online update method keeps changing the location of the centers with any small or large scale change to the power system configuration or operating condition. If this happens continuously (a possibility in a multimachine power system), then the neural network keeps forgetting the previously learned information. In other words, this technique moves the system away from the global minima towards local minima,
- The offline initialization on the other hand, does not allow the neural network to respond accurately to the changes to the power system. Therefore, the network estimates will not closely follow the actual target values.

A third technique is proposed in this chapter for updating the RBF centers that incorporates the benefits of both previous methods. At first, several small scale and large scale disturbances are applied to the power system in a way that they can provide a true representation of the system dynamics. In a real power system where applying faults to the system is not feasible, this large amount of data can be gathered by sampling the system performance over a long period of time. Using the large amount of data samples

of the problem space, the centers are first derived using an offline batch-mode clustering technique (see Appendix C). During the operation of the network, a quasi-online training algorithm is applied for updating the centers: In a window of 50 samples (1 sec), the distances of the RBF centers with the newly arrived data is compared for each center. If for any of the centers this difference is larger than a user defined tolerance level η, then the center is marked as a *potential* center for updating. At the end of the time window, the five centers that show the worst performance, i.e., largest overall distance with the new data, are updated. The centers are then fixed until the end of the next evaluation. The number of centers to be updated depends on the problem and the performance of the neural network.

The output weight matrix of the RBF network undergoes continuous online training in order to be able to respond to minor changes and deviations in the power network.

7.6. Missing Sensor Restoration

In real world problems, the inputs to the neural network come from sensors that can be located far from the network and are prone to unexpected failures or malfunctions. Neural networks are generally considered to be robust to inaccurate data or the partial loss of information of one or more inputs. However, this natural robustness is the result of interdependence between the inputs to the network which provides a partial redundancy (Armitage and Lo, 1994), where some inputs serve as backup information for other network inputs. Clearly, greater levels of robustness can be achieved by introducing more redundant inputs to the network. Therefore, a network of minimal size is not necessarily robust to missing data. Partial robustness can normally be achieved at the cost of greater computational complexity.

Several techniques exist in the literature to deal with the problem of missing sensor restoration. The most common technique used for estimating the unavailable states of the system is state estimation, in which the known states of the system are used to estimate the missing data through direct query of the system model. However, as explained in Section 7.2, this technique has some disadvantages associated with it. Therefore, alternative ways have been employed in the literature in order to overcome this problem without any need for a mathematical model of the plant.

Traditionally, the missing data at the input of the neural network is considered to be zero; however, this can drastically impair the performance of the network. Armitage and Lo (1994) built on the work of Ahmad and Tresp (1993) and proposed a matrix of redundancy coefficients for each pair of inputs in the neural network. The missing input can then be estimated from the set of available inputs using the redundancy coefficients (Armitage and Lo, 1994). Narayanan *et al.* (2002) further extended the previous work by proposing an autoassociative regression neural network (also referred to as *autoencoder*) that learns the relationship among sensor outputs. This method is applied for data restoration in this chapter.

7.6.1. Autoencoder Structure

The autoencoder is a MLP neural network with fewer neurons in the hidden layer than in the input layer, which is trained offline to learn an identity mapping between its inputs and outputs (Figure 7.5). The data runs through a bottleneck that reduces the degrees of freedom among the sensor data. Supervised learning is applied for training the autoencoder. Convergence of the weights of the autoencoder means that the input set can

be reconstructed from the reduced data set and an empirical constraint of the data set has been discovered (Narayanan *et al.*, 2002).

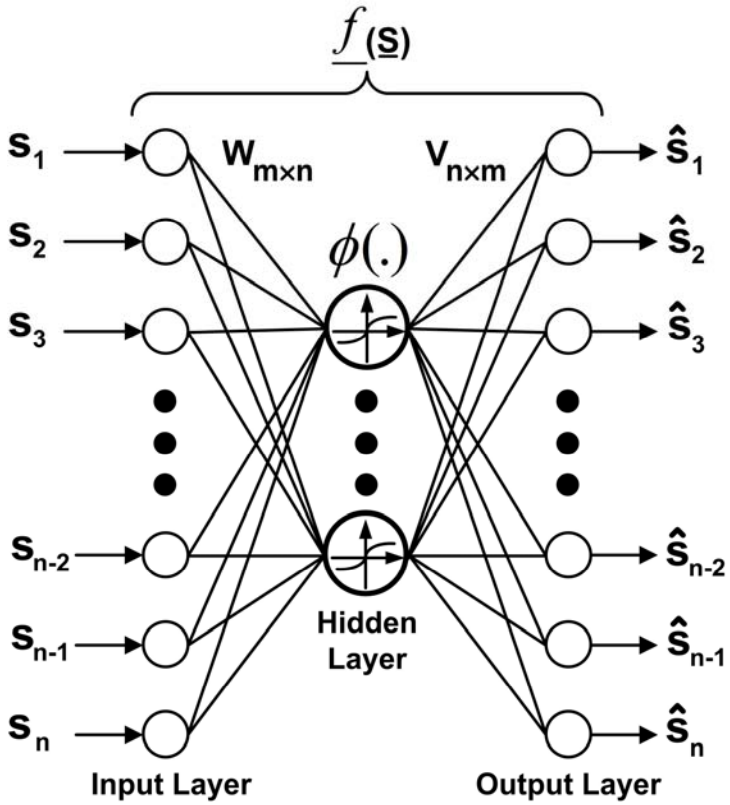

Figure 7.5. Autoencoder structure.

The input to the autoencoder consists of all the sensor readings across the power network. All the plant states and the measurements along with their two times delayed values are fed into the autoencoder. The input vector of the autoencoder can therefore be expressed as:

$$S = [X(t)^T, X(t-1)^T, X(t-2)^T, Z(t)^T, Z(t-1)^T, Z(t-2)^T]^T . \quad (7\text{-}8)$$

With 45 inputs and 45 outputs, the number of neurons in the hidden layer of the autoencoder is heuristically chosen to be 10. This means that the neural network reduces

a 45-dimensional input space to a 10-dimensional input space, and restores the original data from the reduced set. The hyperbolic tangent function is used as the activation function of the autoencoder and backpropagation is used for updating the synaptic weight matrices *W* and *V*.

7.6.2. Projection onto Convex Sets

In general, the autoencoder learns to perform a projection operation onto a subspace spanned by the training data. Narayanan *et al.* (2002) proposed an approach based on projection onto convex sets (POCS) for restoring the missing data. In this technique, the outputs of the autoencoder corresponding to the lost data are iteratively fed back to the autoencoder inputs until the solution converges.

A perfectly trained autoencoder will reproduce the exact input vector at the output layer as:

$$\underline{S} = \underline{f}(\underline{S}), \tag{7-9}$$

which means that the operator is nonexpansive[2]. Therefore the operation will converge to a fixed point. However, this point is not unique and depends on the initialization. Thompson *et al.* experimentally showed that a trained autoencoder is neither strictly contractive nor strictly nonexpansive; instead it is nearly nonexpansive (Thompson *et al.*, 2003). Moreover, they showed that if the initial input vector is bound within the unit cube in *n* dimensions, then the convergence of the iterative process is largely insensitive to the actual value of the operating point and the solution will therefore converge to a unique

[2]A contractive mapping is defined as a mapping $f : X \to X$ in which the output distance between two points is less than the input distance, i.e., $\| f(x) - f(y) \| \leq k . \| x - y \|$, where $0 \leq k < 1$. If $k = 1$ then the mapping is nonexpansive.

point. The fewer sensors that are missing, the more likely the autoencoder will have a unique value of convergence (Thompson *et al.*, 2003).

The structure of the overall sensor restoration is shown in Figure 7.6. The output of the autoencoder associated with the missing sensors is fed back to its input. The portion of the output corresponding to the healthy sensor readings is ignored.

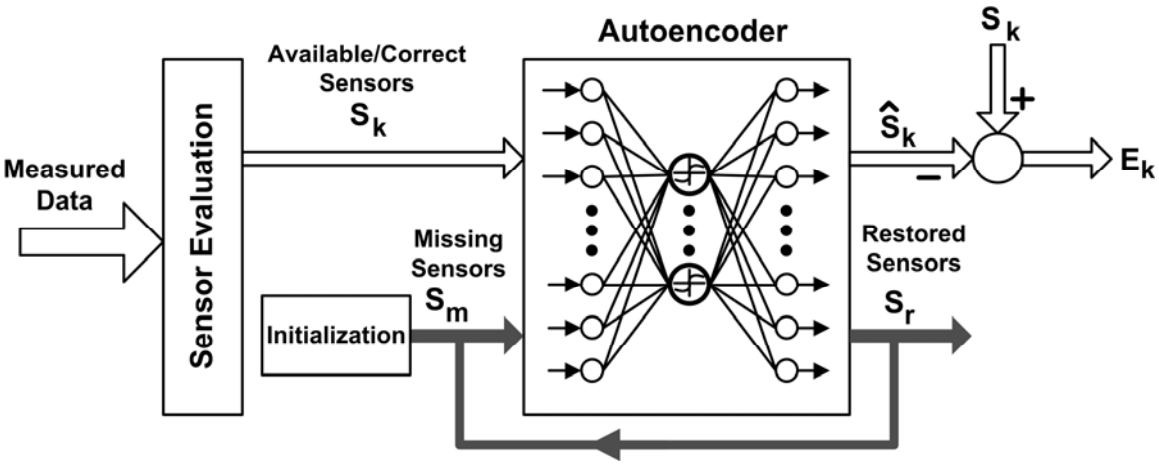

Figure 7.6. Missing sensor restoration structure.

This process is iterated back and forth until a certain stopping criterion is reached. There should be a maximum number of iterations defined for the process to ensure that a final estimate for the missing sensors will be ready for the neuroidentifier on time. Moreover, the error between the correct sensor readings and their estimates in the output of the autoencoder should be closely watched, i.e., the signal E_k in Figure 7.6. An increase in this error will indicate that the iteration process has diverged and the final results should not be used for the neuroidentifier.

7.7. Simulation Results

Several cases are studied in this chapter in order to validate the effectiveness of the proposed WASP for estimating the states of the power system in real-time. The case studies are classified into two categories: cases with static transport lag and cases with dynamic transport lag.

7.7.1. Static Transport Lag, $T_L < 100ms$

7.7.1.1. Case Study 1: No Changes in the System Configuration

7.7.1.1.1. Three Phase Short Circuits

In the first case, the power system is exposed to several large scale disturbances that do not cause any configuration changes to the system. It is considered that the vectors of states/measurements are available at $(t-100ms)$. The WASP estimates the states of the plant based on those values. The actual values of the power system states at time (*t*), i.e., the speed deviations of the synchronous generators, are compared with their corresponding estimates by the WASP at the same time step. The actual values of the states at time $(t-100ms)$ are also shown in the following simulation results.

In the first test, a 100 *ms* three phase short circuit is applied to the middle of the transmission line connecting buses 2 and 5 after 12 sec. The line is considered to remain in the circuit after the fault is cleared. Figure 7.7 shows the speed deviations of the second generator. The same state variable is shown in Figure 7.8 while the system is exposed to a 100 *ms* three phase short circuit at the middle of the transmission line connecting buses 7 and 8. Once again the estimated value at time (*t*) follows the corresponding actual value with a good precision.

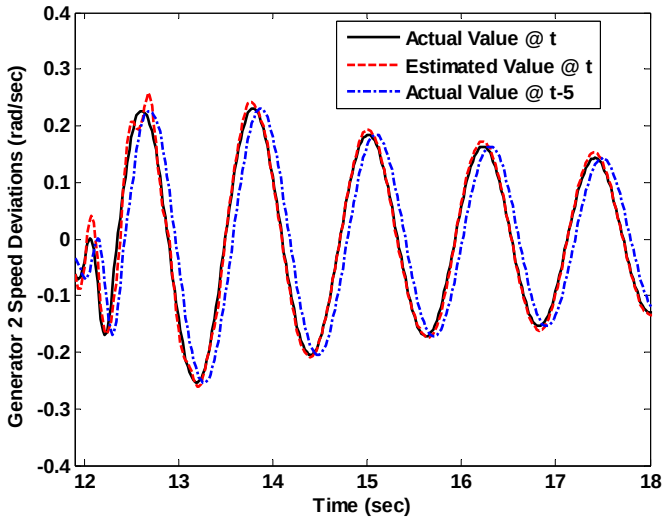

Figure 7.7. Actual and estimated speed deviations of generator 2 during a three phase short circuit at the transmission line connecting buses 2 and 5.

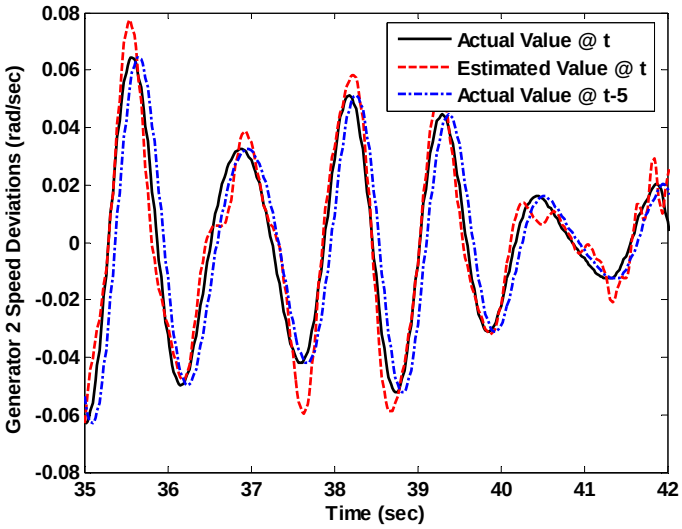

Figure 7.8. Actual and estimated speed deviations of generator 2 during a three phase short circuit at the transmission line connecting buses 7 and 8.

Figure 7.9 depicts the speed deviations of generator 3 during another test where a 100 *ms* three phase short circuit is now applied to the middle of the transmission line connecting buses 3 and 4.

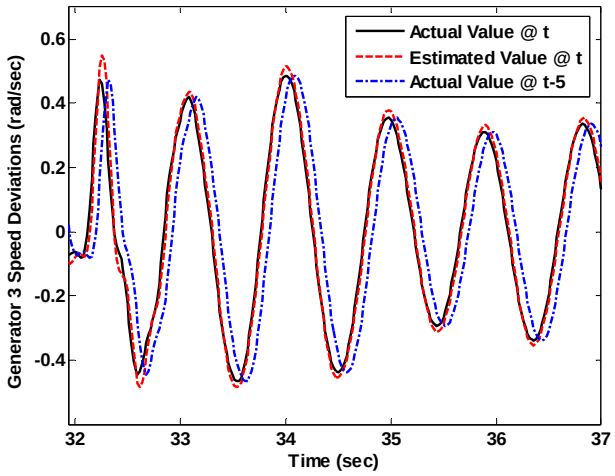

Figure 7.9. Actual and estimated speed deviations of generator 3 during a three phase short circuit at the transmission line connecting buses 3 and 4.

7.7.1.1.2. Small PRBS Disturbances

The system is also tested by applying PRBS disturbances to its inputs. Figure 7.10 shows some typical results where PRBS disturbance is applied to the line voltage reference of the STATCOM. The magnitude of the PRBS is limited to $\pm 5\%$ of the steady state voltage reference, while its frequency is a combination of 0.5, 1 and 2 Hz.

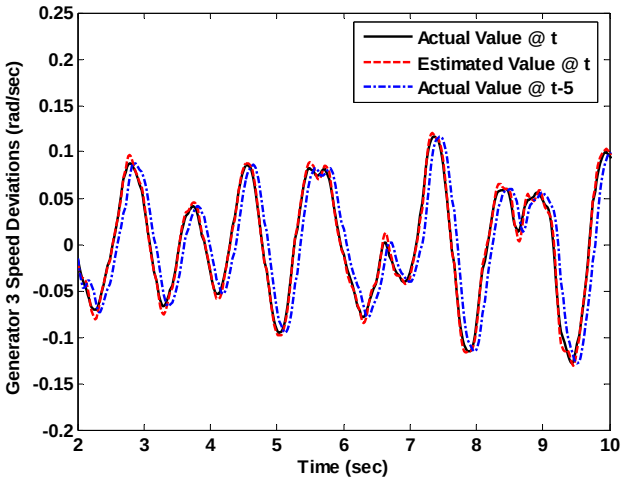

Figure 7.10. Actual and estimated speed deviations of generator 3 during PRBS disturbance applied to the STATCOM voltage reference.

7.7.1.2. Case Study 2: Power System Configuration Change

It was assumed in the previous case study that the power system configuration does not change as a result of the disturbances applied to the network. This is not necessarily true in a real life power grid. Various faults and disturbances can force a transmission line or a shunt load to be removed from the system even after the fault has been cleared. In this section, several simulation results are provided which correspond to the disturbances that force the removal of a line.

In the first test, one of the transmission lines connecting buses 3 and 4 is disconnected. Figure 7.11 shows a close up view of the actual and estimated speed deviations of generator 3.

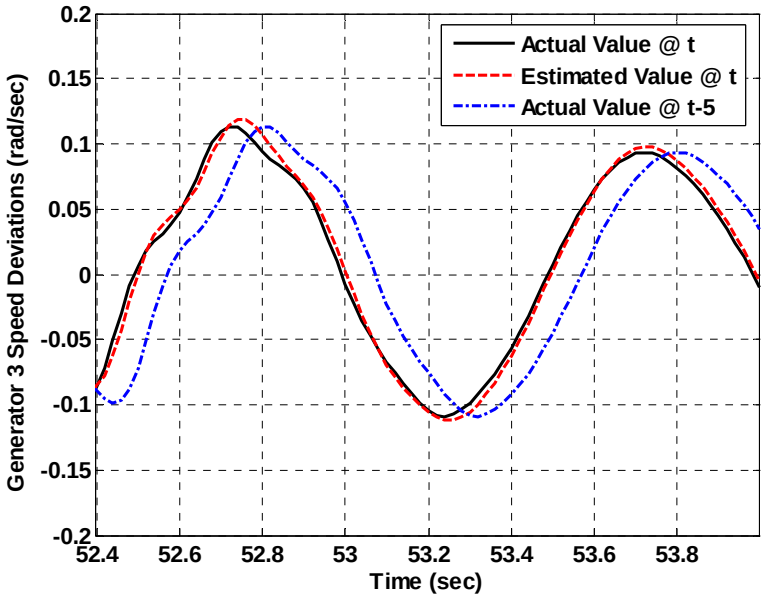

Figure 7.11. Actual and estimated speed deviations of generator 3 when one of the parallel lines connecting buses 3 and 4 is disconnected.

In another test, the system is exposed to a 100 *ms* three phase short circuit at one of the parallel lines connecting buses 3 and 4. The line is disconnected after the fault is cleared. Figure 7.12 shows the speed deviations of generator 4.

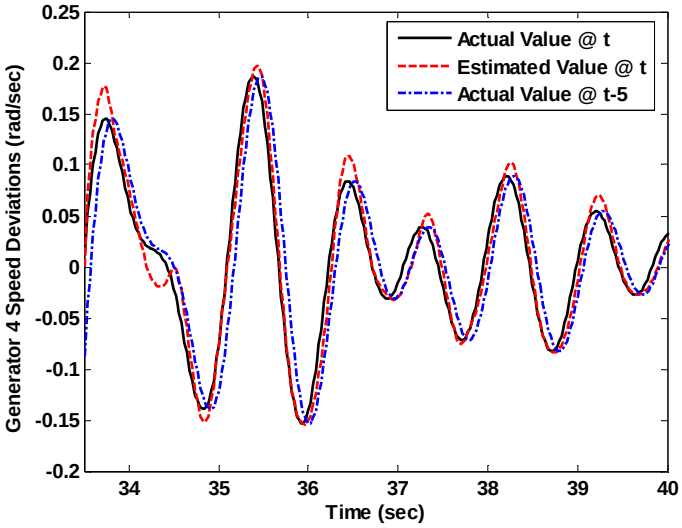

Figure 7.12. Actual and estimated speed deviations of generator 4 during a three phase short circuit at one of the transmission lines connecting buses 3 and 4. The line is removed after the fault is cleared.

7.7.1.3. Case Study 3: Power System and State/Measurement Vector Configuration Change

All the results shown in the previous section were related to cases where the change in the power system configuration does not affect the measurement process of any of the state variables or the plant measurements. However, in general, changes to the system configuration might at times cause the loss of one or more state/measurement variables. This can happen for instance when a transmission line whose active power flow is considered to be one of the measured variables is disconnected as a result of a fault. This

specific input data to the neuroidentifier will be unavailable before the line is restored, maybe even after a few hours. While the neuroidentifier should store the previously learned information for the time when the line is restored, it should be able to temporarily deal with the current situation.

One way to compensate for this problem is to remove the information related to the lost input from the input of the RBF network in Figure 7.4 and its corresponding connections. This approach is referred to as *pruning technique* and is normally used for downsizing a neural network in order to obtain the optimum number of neurons in the hidden layer (Reed, 1993). Nevertheless, the idea can also be employed for the WASP with lost information.

Consider the case where the k^{th} sensor measurement is lost. Suppose this data along with its the corresponding time delayed values appear in the r^{th}, $(r+1)^{th}$ and $(r+2)^{th}$ input of the RBF network in Figure 7.4. Given the fact that the network has m inputs and n neurons in the hidden layer, the pruning analysis will remove the r^{th}, $(r+1)^{th}$ and $(r+2)^{th}$ inputs from the RBF input vector reducing it to a $(m-3)\times 1$ column vector. Moreover, it will omit the corresponding r^{th}, $(r+1)^{th}$ and $(r+2)^{th}$ columns from the $n \times m$ matrix of the RBF centers. Clearly, the neuroidentifier centers and output weight matrix still undergo training (as explained in section 6.4.1) so that the network adjusts itself to the dynamics of the new system.

In order to examine the robustness of the WASP, a 100 *ms* three phase short circuit is now applied to the middle of one of the transmission lines connecting buses 4 and 6 after 42 seconds. The transmission line is disconnected and the line power P_{46} measurement is lost as a result of the fault. The WASP undergoes pruning and removes the rows/columns

of the center/weight matrix that are related to P_{46}. Figures 7.13-7.15 indicate that the WASP can still manage to estimate the states of the system with an acceptable accuracy. Similar tests applied to various buses and transmission lines show similar results.

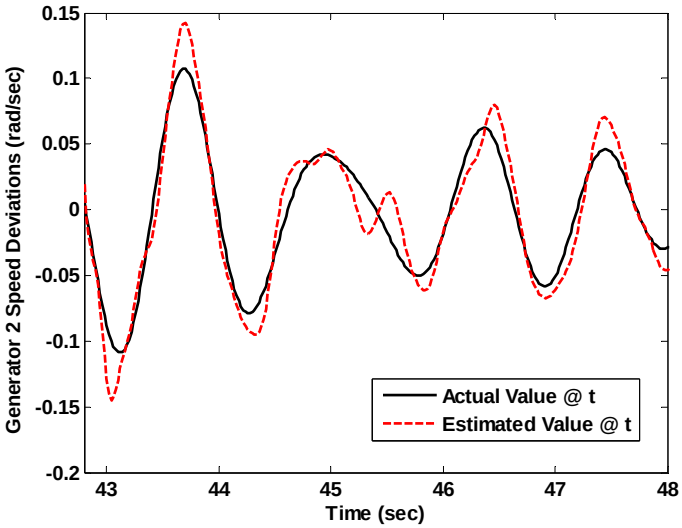

Figure 7.13. Actual and estimated speed deviations of generator 2 during a three phase short circuit at the transmission line connecting buses 4 and 6. The line is removed after the fault is cleared.

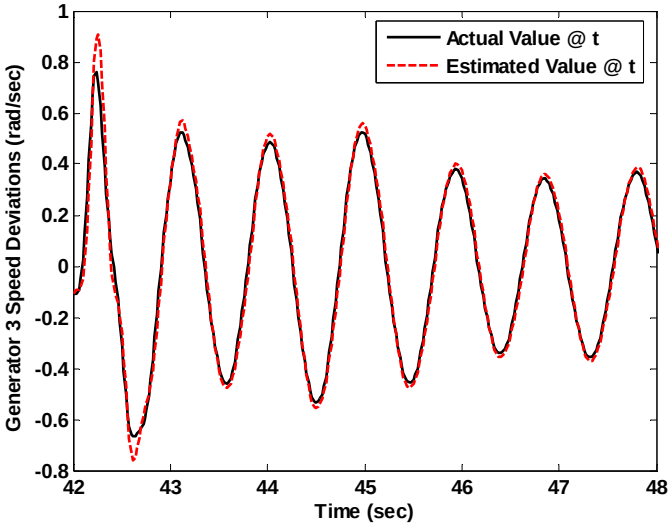

Figure 7.14. Actual and estimated speed deviations of generator 3 during a three phase short circuit at the transmission line connecting buses 4 and 6. The line is removed after the fault is cleared.

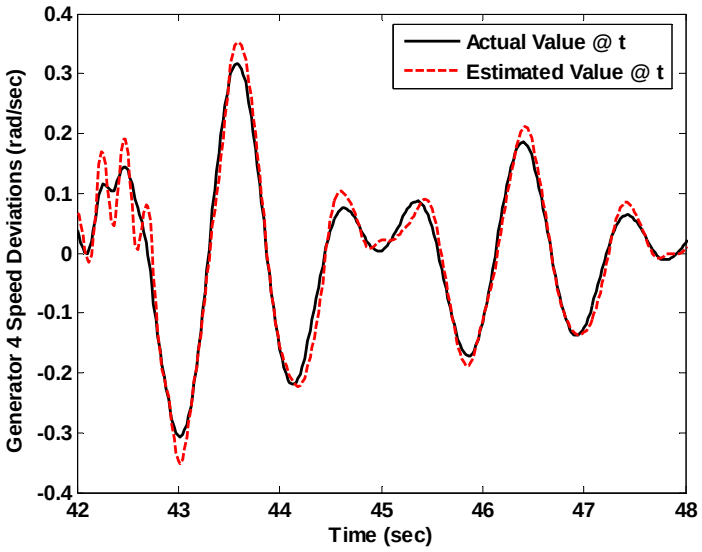

Figure 7.15. Actual and estimated speed deviations of generator 4 during a three phase short circuit at the transmission line connecting buses 4 and 6. The line is removed after the fault is cleared.

7.7.2. Dynamic Transport Lag

7.7.2.1. Case Study 1: Transport Lag 100 *ms* < T_L < 150 *ms*

Although 100 *ms* was considered to be a reasonable threshold for the information to reach the WASP, there can be cases where the data backlog or overused communication channels force the transport lag to go slightly beyond 100 *ms*. One solution can be to consider the unavailable readings as missing data and using the autoencoder to estimate the actual values. However, sometimes it might be more desirable to reduce the number of calculations. Therefore, an alternative solution can be to use the last available data in lieu of the unavailable data, as long as the last available value does not go beyond a second user defined threshold. This threshold is considered to be 150 *ms* in this study, which sounds reasonable for the range of the natural frequencies of oscillation of the power system.

A 100 *ms* three phase short circuit is applied to the transmission line connecting buses 2 and 5. Two conditions are considered: a case where the active power flow P_{25} reaches the WASP with 150 *ms* delay, and a second case where all the measurements coming from generator 2 and its neighborhood, i.e., $P_{25}, \Delta\omega_2$ and V_2 reach the WASP with 150 *ms* delay. Figures 7.16-7.18 show the estimated speed deviations of the synchronous generators by the WASP, substituting the unavailable data at $(t-100ms)$ with the available data at $(t-150ms)$. When one or more generator speed signals are not available, the error correction path shown in Figure 7.2 cannot be performed.

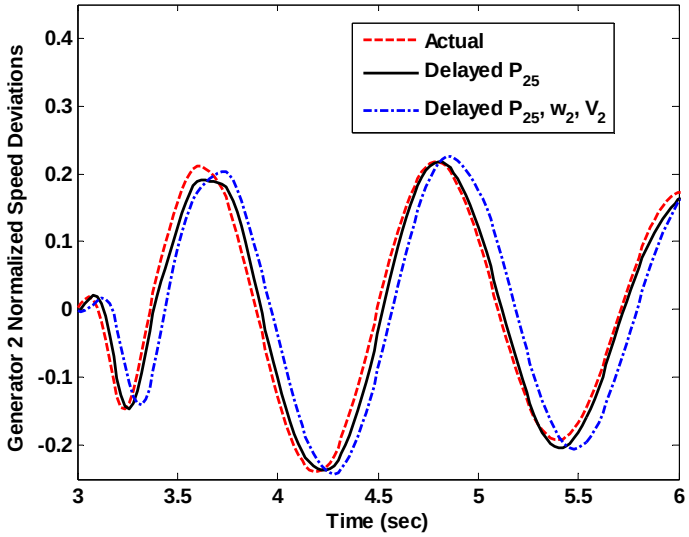

Figure 7.16. Actual and estimated speed deviations of generator 2 during a three phase short circuit at the transmission line connecting buses 2 and 5.

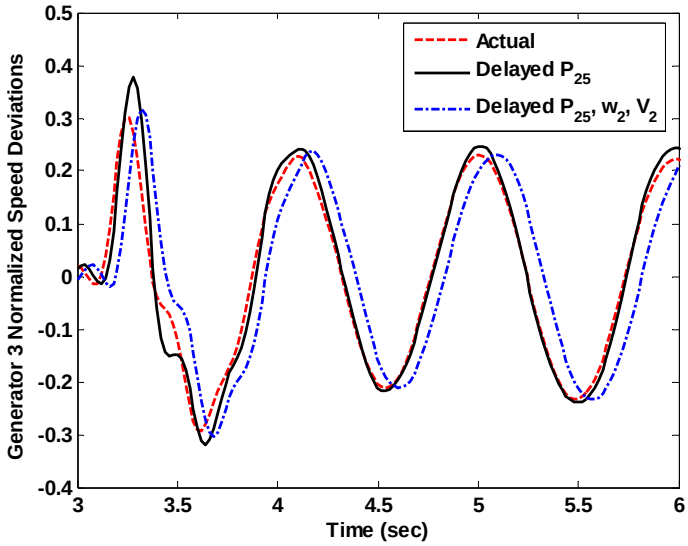

Figure 7.17. Actual and estimated speed deviations of generator 3 during a three phase short circuit at the transmission line connecting buses 2 and 5.

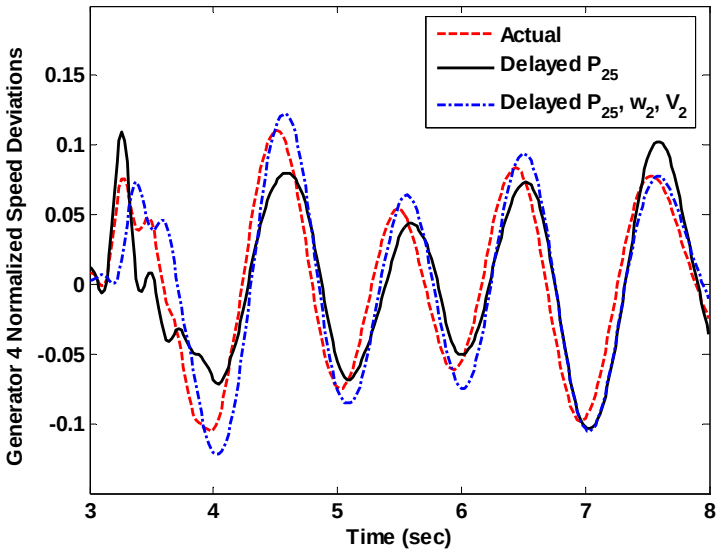

Figure 7.18. Actual and estimated speed deviations of generator 4 during a three phase short circuit at the transmission line connecting buses 2 and 5.

The results show that when only one signal is delayed, the estimated values provided by the WASP are sufficiently accurate. The performance is slightly affected when more

signals are over-delayed[3] by 150 *ms*. In this case, the WASP cannot estimate the states in real time, instead the estimated outputs of the WASP are delaying the corresponding actual plant states by almost 70 *ms*. In order to analytically evaluate the efficiency of the WASP, a performance index *P.I.* is defined as:

$$P.I. = \left[\frac{1}{N} \sum_{i=1}^{N} \left(\sum_{j} (\Delta\omega_{act,j}^{i} - \Delta\omega_{est,j}^{i})^2 \right) \right]^{-1}, \tag{7-9}$$

where $\Delta\omega_{act,j}^{i}$ and $\Delta\omega_{est,j}^{i}$ are the actual and estimated values of the j^{th} generator speed deviations corresponding to the i^{th} sample, and N is the total number of samples. A higher performance index *P.I.* indicates more accurate state estimation by the WASP. Table 7.1 summarizes the results for three cases: without the WASP and using the available value at $(t - 150ms)$ in lieu of the desired real-time value, with the WASP and one delayed measuement, with the WASP and three delayed measurements.

Table 7.1. WASP performance index for over-delayed measurements.

Quantity	No WASP	WASP with one over-delayed input	WASP with three over-delayed inputs
Estimation error[4] for $\Delta\omega_2$	0.0066	0.000453	0.0018
Estimation error for $\Delta\omega_3$	0.0120	0.0010	0.0037
Estimation error for $\Delta\omega_4$	0.0014	0.000425	0.000628
Normalized Performance index	0.096	1.0	0.312

[3] The term "over-delayed" is used in this text to indicate the case where the transport lag is larger than the assumed 100 *ms* and less than a threshold of 150 *ms*.

[4] Estimation error for $\Delta\omega_j$ is defined as: $\frac{1}{N}\sum_{i=1}^{N}(\Delta\omega_{act,j}^{i} - \Delta\omega_{est,j}^{i})^2$

It can be seen that even in the case of having three over-delayed inputs, the WASP can improve the performance index from 0.096 when using the same over-delayed values to 0.312 when using the WASP estimates. This can improve the performance of the controller using those estimates.

7.7.2.2. Case Study 2: Transport Lag $T_L > 150$ *ms*

When the transport lag corresponding to a measurement/state signal is larger than 150 *ms*, the proposed WASP considers that signal to be missing and tries to restore the missing data from the information that is already available. The procedure explained in Section 7.5 for restoring the missing sensors is adopted for this purpose. Various case studies are presented in this section in order to show the efficiency of the proposed technique.

In the first test, it is assumed that the sensor reading the active power flow through the transmission line connecting buses 2 and 5 has failed during a three phase short circuit applied to the line. The autoencoder is activated in order to restore this information from the available data. Figures 7.19-7.21 show the results.

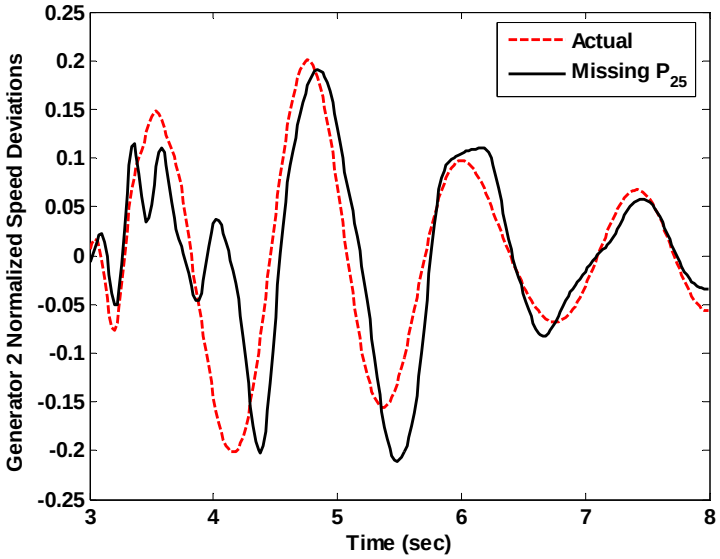

Figure 7.19. Actual and estimated speed deviations of generator 2 during a three phase short circuit at the transmission line connecting buses 2 and 5.

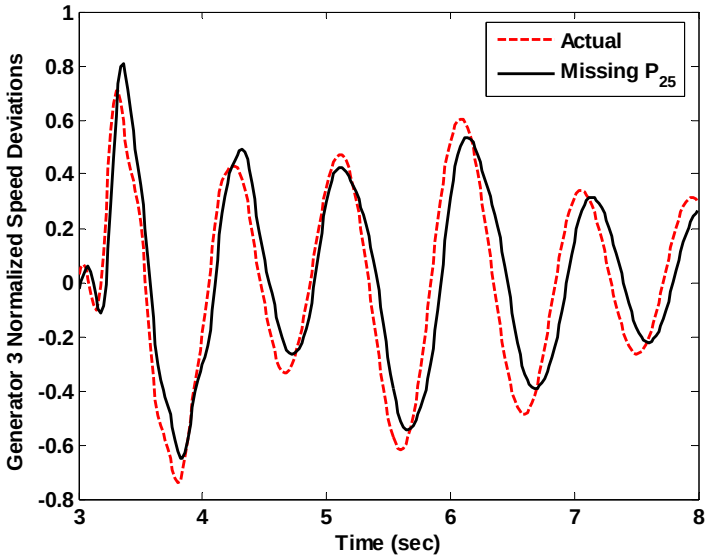

Figure 7.20. Actual and estimated speed deviations of generator 3 during a three phase short circuit at the transmission line connecting buses 2 and 5.

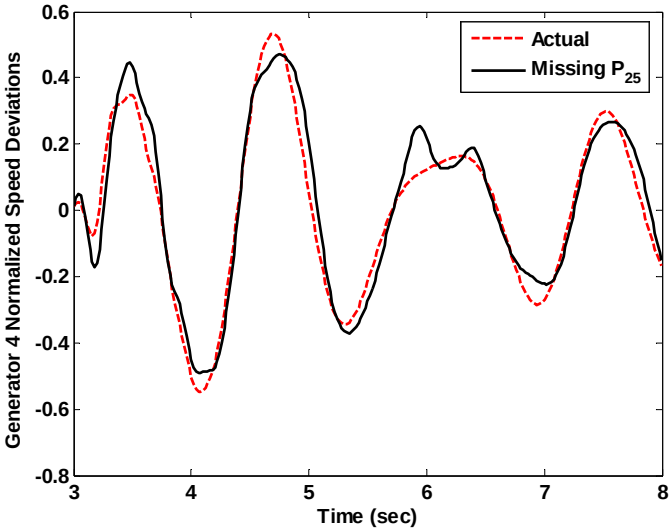

Figure 7.21. Actual and estimated speed deviations of generator 4 during a three phase short circuit at the transmission line connecting buses 2 and 5.

In the second test, a three phase short circuit occurs at the middle of one of the parallel transmission lines during which the sensor that reads the generator 3 speed fails. Figure 7.22 shows the estimated and actual values of the generator speed.

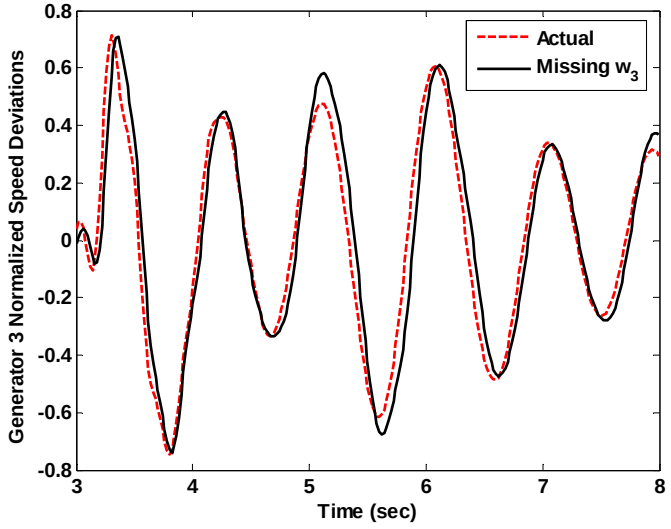

Figure 7.22. Actual and estimated speed deviations of generator 3 during a three phase short circuit at the transmission line connecting buses 2 and 5.

Another test is carried out by assuming that the sensor reading the rotor speed of generator 4 has failed. Figure 7.23 shows some typical results.

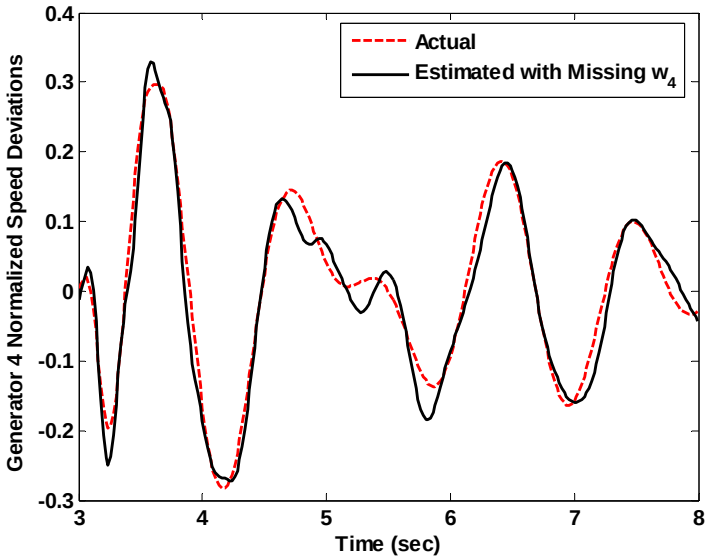

Figure 7.23. Actual and estimated speed deviations of generator 4 during a three phase short circuit at the transmission line connecting buses 2 and 5.

7.8. Computational Intensity of the WASP

In this section the computational intensity associated with the WASP is analyzed. The worst case scenario is considered where the data is missing and the autoencoder is used for estimating the missing data. Moreover, it is assumed that all these calculations coincide with the specific time step where the RBF unit centers need to be updated. Table 7.2 summarizes the computational intensity. Parameters m, n and r denote the number of neurons in the input, hidden and output layers respectively. Index a corresponds to the autoencoder, while index i indicates the neuroidentifier. A maximum number of iterations of k is assumed for the autoencoder estimation loop. The last row in Table 7.2 gives a

numerical example for the special case of the WASP shown in Figure 7.3 with k considered to be 10.

Table 7.2. Worst case analysis of the number of calculations required by the proposed WASP.

Step	$+/-$	$\times/\div$	$e^{(.)}$	$\sqrt{\ }$
Feedforward path of the autoencoder	$k_a[n_a.(m_a+r_a+1)-r_a]$	$k_a[n_a.(m_a+r_a+1)]$	n_a	0
Feedforward path of RBFN #1	$n_i.(m_i+r_i)-r_i$	$n_i.(m_i+r_i+2)$	n_i	0
Backpropagation path of RBFN #1	$r_i.(2n_i+1)$	$3.r_i.n_i$	0	0
Center update of RBFN #1	$m_i.(n_i^2+n_i+2)+n_i$	$m_i.(n_i^2+n_i+1)+m_i$	0	$2n_i$
Feedforward path of RBFN #2	$n_i.(m_i+r_i)-r_i$	$n_i.(m_i+r_i+2)$	n_i	0
Computational intensity of the propsed WASP (Figure 7.3)	94533	95199	110	100

Since the time step of 20 ms is selected for the WASP (see Section 7.5.1), any processor designed for implementing the WASP should be able to complete the above number of calculation within one time step of the WASP performance, i.e., in less than 20 *ms*.

7.9. Conclusions

Any supervisory level controller, designed for a power system, requires access to the network data in real-time. Clearly, the transport lag associated with the data transmission within the power system causes the data to reach the controller with delay. This can

create problems in a plant with fast dynamics, such as a multimachine power system, where the control decision is highly dependent on the data accuracy.

A neural network based wide area state predictor (WASP) has been proposed in this chapter that estimates the plant states based on the time delayed measurements available. Moreover, by using a neural network based autoencoder, the proposed WASP is able to restore the missing data from the available data in case of having failed sensors or more than usual delays.

Various case studies have been presented that indicate the WASP is able to estimate the plant states in real-time with good accuracy at various operating conditions and system contingencies. Such a scheme can be effectively used in any wide area or supervisory level controller in a power system.

CHAPTER 8

ADAPTIVE CRITIC DESIGNS BASED NEURO-FUZZY EXTERNAL CONTROLLER FOR A STATCOM

8.1. Introduction

The previous chapters focused on designing intelligent adaptive and/or optimal controllers for a STATCOM in a multimachine power system. Artificial neural networks and fuzzy logic based systems were applied in order to achieve this. The main objective of the STATCOM in all those studies was to control the voltage at the point of connection to the power network.

However, it has been widely shown that controllers can be designed for a STATCOM that can also enable it to improve the dynamic and/or transient stability of the power system (Mathur and Karma, 2002; Sun *et al.*, 2002; Yang *et al.*, 2001; Valderrama *et al.*, 2001; Chen and Joós, 2001; Hingorani and Gyugyi, 1999; Patil *et al.*, 1998; Han *et al.*, 1998; Lehn and Iravani, 1998). These capabilities can also be achieved as peripheral control objectives for the device, by introducing auxiliary signals to the STATCOM controller (Hingorani and Gyugyi, 1999). This concept is referred to in this chapter as *external control*, since network signals, other than the local measurement signal, i.e., the line voltage, are used to bias the STATCOM line voltage reference (Figure 2.6).

Like all other engineering techniques, a simplified linear method can be applied for implementing the external controller. However, trial and error based search techniques are normally employed in order to derive the corresponding gains and possibly time constants of the external control path. Clearly, this will not necessarily lead to an

optimum performance. Moreover, such a design will be dependent on the operating condition of the power system which changes continuously as the transmission lines and shunt loads are switched on/off. Therefore, having a nonlinear adaptive or robust controller seems to be the most efficient solution.

In this chapter an optimal neural network based fuzzy (*neuro-fuzzy*) external controller is designed for a STATCOM placed in the IEEE 12-bus benchmark system (Jinag *et al.*, 2006). This neuro-fuzzy external controller is implemented based on the adaptive critic designs (ACD) theory and provides an auxiliary input for the STATCOM internal controller in the form of an additional voltage reference signal. This external controller is optimal and ensures that the STATCOM improves the transient and/or dynamic behavior of the power system, by controlling the network parameters in its neighborhood. In this study, the STATCOM internal controller is assumed to be a conventional PI type which is fine tuned at a single system operating condition. Any of the adaptive/optimal internal controllers discussed in Chapters 5 and 6 can also be used for the STATCOM internal controller; however, for simplicity and without loss of generality, a common PI controller is employed.

8.2. STATCOM External Control

8.2.1. Basic Idea

Figure 8.1 illustrates the IEEE 12-bus benchmark system with a STATCOM connected to bus 4 to provide extra voltage support. In addition to the voltages at buses 4 and 5, several other parameters across the network are directly or indirectly affected by the STATCOM. However, not all of them can be effectively controlled by the STATCOM.

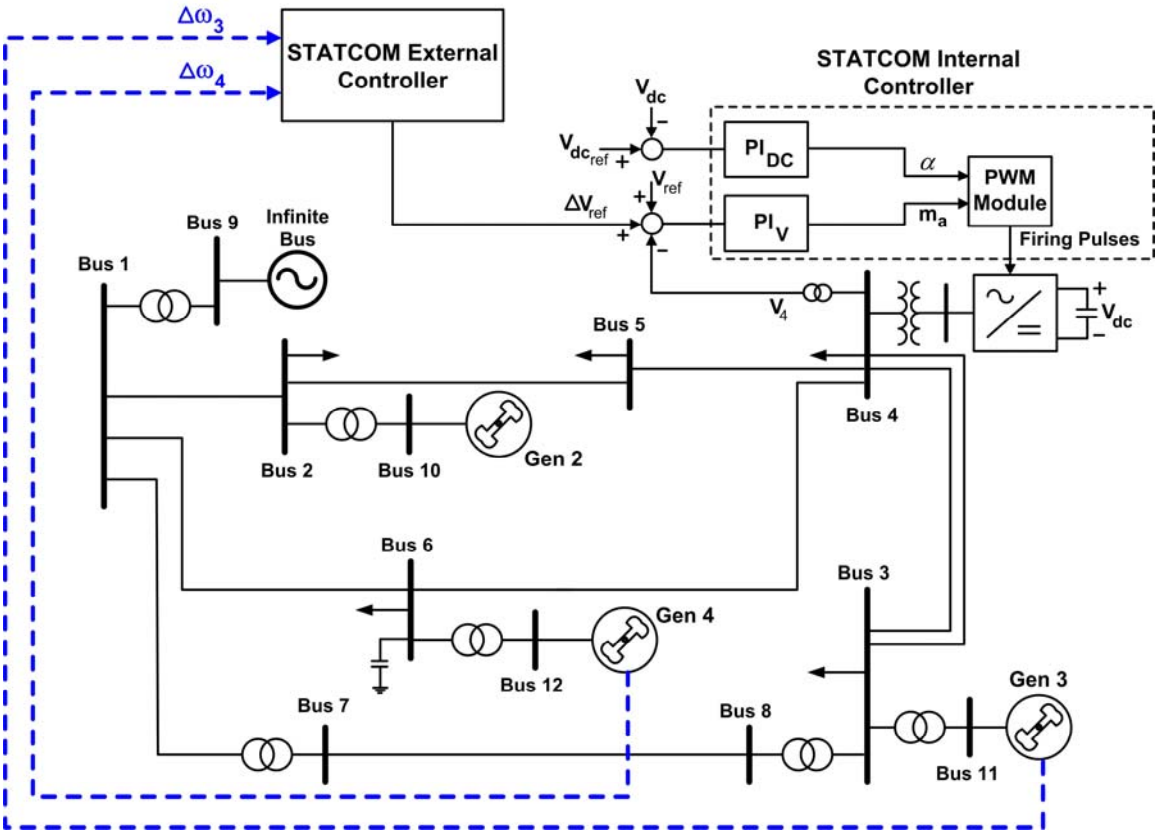

Figure 8.1. STATCOM external controller.

Since the primary effect of the STATCOM on the power system is to regulate the voltage magnitude and angle at the point of common coupling (PCC), active power flows through the neighboring transmission lines and the speed deviations of the synchronous generators are the main parameters that are directly affected by the STATCOM. It was observed in this study that among the three synchronous generators in Figure 8.1 the speed deviations of synchronous generators 3 and 4, i.e., $\Delta\omega_3$ and $\Delta\omega_4$ are affected most and are therefore considered as the primary variables of interest. Controlling the two speed deviations will result in damping out the low frequency power oscillations in the transmission lines connecting the STATCOM to the synchronous generators, i.e., P_{34}

and P_{46}, as well as other transmission lines neighboring the two generators. Generator 2 is considerably closer to the infinite bus and therefore has substantial damping present in its speed deviations, and in the power flow fluctuations of the neighboring transmission lines. The STATCOM can therefore only slightly improve the system damping around this generator.

It is known that an active Var compensator with suitable controller can improve the dynamic stability of the power system it is connected to (Miller, 1982). In its simplest form, this can be a bang-bang control, where the compensator injects/withdraws reactive power into the network in response to the oscillations in the active power (or load angle).

During the time that the transmission line active power and the load angle are increasing, reactive power injection into the network causes an increase in the PCC voltage which opposes the change in the active power and the load angle. Similarly, a decrease in the active power and load angle can be compensated by withdrawing reactive power from the network and therefore decreasing the voltage (see Section 2.4.3). Although this control procedure is effective in damping out the line active power flow and the load angle variations, it causes voltage deviations at the network buses. However, during large scale disturbances, where maintaining the system stability is of the highest priority, these voltage fluctuations can be ignored. Although care should be taken that at none of the buses the voltage falls outside the acceptable range, which is normally [0.95,1.05] p.u.

Since measuring and analyzing the speed deviations of the generators are easier and more practical than the line active power flows or the generator load angle, $\Delta\omega_3$ and $\Delta\omega_4$ are selected as the main inputs to the external controller. An increase in the load

angle δ of any generator (and therefore the active power flow of the neighboring lines) can be explained in terms of an increase in the corresponding generator's speed with respect to the steady state rotor speed and vice versa.

8.2.2. Linear External Controller for the STATCOM

Generators 3 and 4 have inertia constants of 3.0 and 5.0 MW.s/MVA respectively, which result in the local swings with frequencies of approximately 1 Hz for generator 3 and 0.8 Hz for generator 4. The fact that the two machines oscillate with different frequencies complicates the external control scheme, since the auxiliary control signal which is suitable for generator 3 for example, might at times exacerbate the dynamic oscillations of the rotor of generator 4 and vice versa.

Figure 8.2 illustrates the three conventional external control schemes that are investigated and compared in this section:

- *Linear controller 1*: external damping controller for the STATCOM to counteract the rotor oscillations $\Delta\omega_3$ in generator 3. The output of the controller, i.e., ΔV_1, is applied to the line voltage reference of the STATCOM,
- *Linear controller 2*: external damping controller for the STATCOM to counteract the rotor oscillations $\Delta\omega_4$ in generator 4. Similar to the linear controller 1, the output of this controller, i.e., ΔV_2, is applied to the line voltage reference of the STATCOM,
- *Linear controller 3*: consists of the linear controllers 1 and 2 in parallel. The gains and time constants of the two paths of controller 3 are the same as the corresponding linear controllers 1 and 2. The total output of the controller, i.e., ΔV_3, is now used as the auxiliary reference signal for the STATCOM.

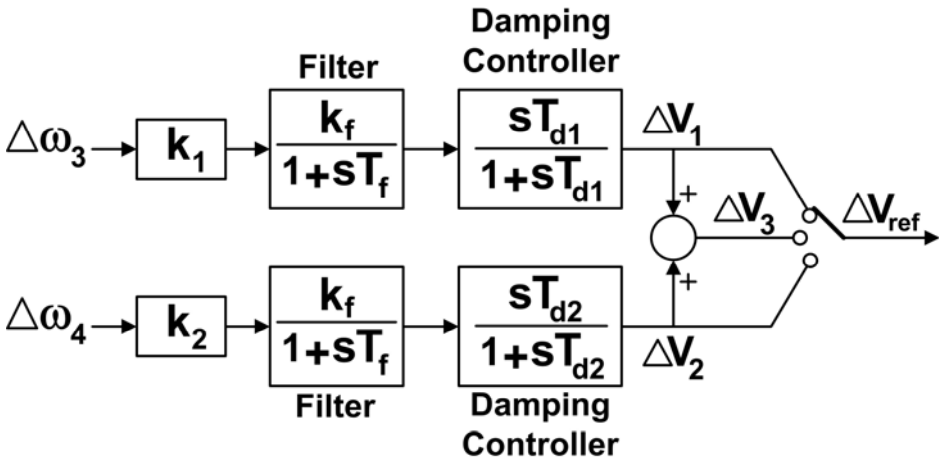

Figure 8.2. Linear (conventional) external controller for the STATCOM.

These conventional controllers are fine tuned and their performances are compared with one another for a selected number of natural faults that typically occur in the power system in Figure 8.2. The effectiveness of the controllers is then compared by defining a performance index as a function of the deviations of the rotor speeds in the three generators. It was observed in the simulations that the STATCOM external controller does not have major impact on the rotor speed deviations of generator 2; nevertheless, it is included in the performance index in Equation (8-1):

$$P.I_i = \left(\sqrt{\frac{1}{N}\sum_{k=1}^{N} \Delta\omega_{2,k}^2} + \sqrt{\frac{1}{N}\sum_{k=1}^{N} \Delta\omega_{3,k}^2} + \sqrt{\frac{1}{N}\sum_{k=1}^{N} \Delta\omega_{4,k}^2} \right)^{-1}, \qquad (8\text{-}1)$$

where $\Delta\omega_{j,k}$ represents the k^{th} sample of the rotor speed deviations of the j^{th} generator and index i represents the i^{th} test. During each fault/disturbance applied to the system, 100 samples are taken from each rotor speed in 10 sec of simulation. The performance indices of different linear controllers are evaluated during various faults and the final performance index is derived according to Equation (8-2).

$$P.I. = \left(\sum_i \frac{1}{P.I_i} \right)^{-1}, \tag{8-2}$$

Table 8.1 summarizes the performance indices of the conventional external controllers obtained from the simulations. Higher values of the performance index indicate better performances by the corresponding controllers. The performance indices in each row of the table are normalized based on the maximum performance index of the corresponding row.

Table 8.1. Performance indices of the conventional external controllers for the STATCOM.

Type of Test	Normalized Performance Index			
	No External Controller	Linear Controller 1	Linear Controller 2	Linear Controller 3
100 *ms* 3-phase short circuit at transmission line 2-5	0.46	0.93	0.52	1.0
100 *ms* 3-phase short circuit at transmission line 3-4	0.64	0.89	0.66	1.0
100 *ms* 3-phase short circuit at transmission line 4-6	0.68	1.0	0.52	0.85
100 ms 3-phase short circuit at transmission line 4-5	0.57	1.0	0.60	0.98
100 *ms* 3-phase short circuit at transmission line 7-8	0.45	0.72	0.44	1.0
Line 2-5 switch on/off	0.79	0.77	1.0	0.87
Line 4-6 switch on and off	0.45	0.65	0.48	1.0

The results in Table 8.1 rule out the second external control scheme, since it cannot provide considerable positive damping for the two generators. Linear controller 1 has a

reasonably high performance index among the three proposed structures, which indicates the STATCOM has a major impact on the dynamic behavior of generator 3. This is due to the fact that two parallel transmission lines connect the FACTS device to generator 3, which results in a low line impedance between the two. Linear controller 3 proves to have the most efficient overall performance among the three proposed techniques. However, Table 8.1 also shows that this controller does not show a consistent performance during various tests. It has a superior performance during the majority of the test faults, whereas it is outperformed by the other linear controllers during some certain disturbances.

This is due to the fact that the dynamics of the nonlinear power system in Figure 8.1 are too complicated for a conventional controller to be able to perform an effective multiple input single output control. The first challenge is that the dependence between the two rotor speeds is far from linear. In other words, the rotor speed deviations of the two generators do not follow the same trajectory. This causes the linear controllers to lose their effectiveness, since actions that must be taken in order to damp out the rotor oscillations of one generator, are at times in contrast with the appropriate action for damping out the oscillations of the other one. Moreover, the controllers' performances strongly depend on the type of the fault, and operating condition, as illustrated by the second and fourth columns of Table 8.1.

8.2.3. An Alternative: Intelligent External Controller

The results of the conventional controllers studied in the previous section indicate that the level of difficulty to establish a relationship between the inputs and the outputs of the nonlinear non-stationary system in Figure 8.1, is too high for a linear controller. This is specifically causing trouble for the conventional control scheme due to the fact that the

oscillations on generators 3 and 4 are out of phase. This problem can be solved by using an intelligent control scheme, in which the controller exerts a control action while watching its effect on the overall performance of the power system.

An ACD based neuro-fuzzy controller can be an efficient solution for the external control of a STATCOM, since:

- the relatively low number of inputs and outputs (rotor speed of the two generators ω_3 and ω_4 as the controller inputs and the auxiliary reference signal to the STATCOM as the output) allows the creation of an effective rule base for a fuzzy controller;
- the introduction of neural networks enables the controller to perform in an optimal way by evaluating the effects of its control actions on the response of the power system, and updating the controller parameters accordingly.

Applications of connectionist learning systems such as artificial neural networks in adaptively adjusting the parameters of fuzzy systems have been discussed in the literature (Figueiredo and Gomide, 1999; Jang and Sun, 1995; Lin and Lee, 1994; Jang, 1993; Lin and Lee, 1991). In a typical neuro-fuzzy system, the parameters of the fuzzy controller, such as the membership functions and the consequent rules, are considered as the synaptic weights of a connectionist learning system. Neural network based learning techniques are then applied in order to adjust these parameters based on the performance of the system.

In this chapter, a fuzzy logic based controller is designed for external control of the STATCOM in Figure 8.1. An adaptive critic designs based technique is then applied for optimally updating the parameters of the fuzzy controller. The updates are provided by an

action dependent heuristic dynamic programming (ADHDP) based Critic network which is trained to estimate the cost-to-go function associated with the power system performance (see Chapter 6).

8.3. STATCOM Neuro-Fuzzy External Controller

Figure 8.3 shows the schematic diagram of the proposed STATCOM neuro-fuzzy external controller. It is an adaptive critic designs based fuzzy controller which is capable of providing near optimal results in the presence of noise and uncertainty in a nonlinear system, such as the power system shown in Figure 8.1 (see Chapter 6).

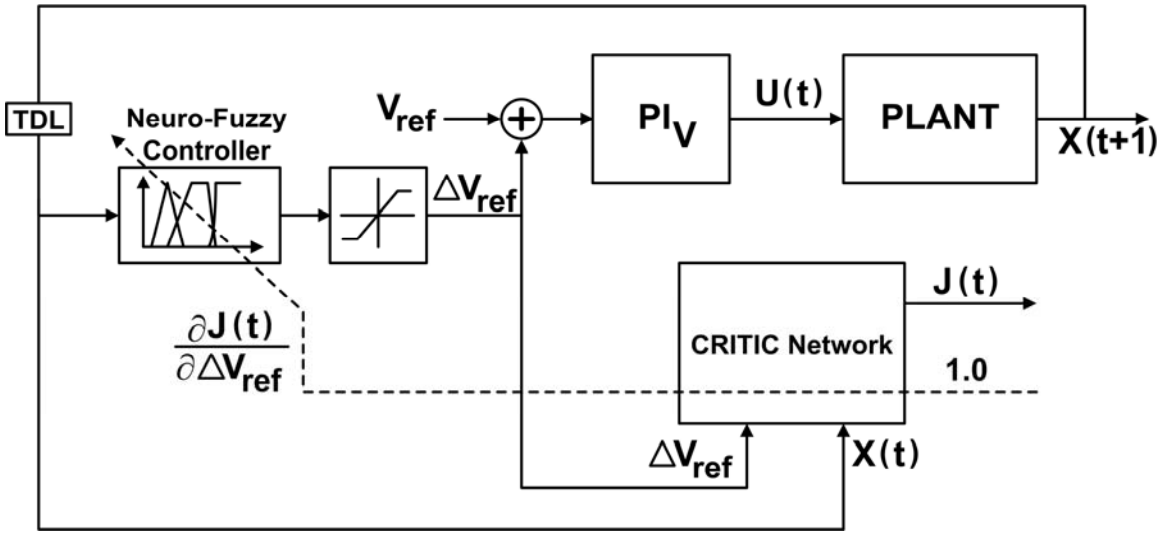

Figure 8.3. Schematic diagram of the STATCOM neuro-fuzzy external controller.

The proposed controller consists of two main components:

- Fuzzy logic based controller, which functions as a nonlinear gain scheduling controller and is trained to send the optimum control signals to the plant, resulting

in minimization/maximization of the function *J* over the time horizon of the problem,

- Critic neural network, which is a neural network trained to approximate the cost-to-go function *J* required for infinite horizon optimization.

8.3.1. Fuzzy Controller

The heart of the neuro-fuzzy controller is the fuzzy inference system. A first order Takagi-Sugeno fuzzy model is used for implementing the controller, which is a special case of the Mamdani model. The input to the fuzzy controller is the vector of the selected states of the power system as in Equation (8-3):

$$X(t) = [\Delta\omega_3(t), \Delta\omega_4(t)]^T, \tag{8-3}$$

The neuro-fuzzy controller in return generates a control signal ΔV_{ref}, which is added to the line voltage reference of the local PI_V controller (Figure 8.3). At steady state, the PI_V has a line voltage reference of 1.0 p.u. Therefore, the output of the neuro-fuzzy controller is clamped at ± 0.05 p.u., such that the voltage at bus 4 does not fall outside the acceptable range of [0.95,1.05] p.u.

Five membership functions are considered for the rotor speed deviations of each generator, which are associated with the fuzzy terms *Negative Big*, *Negative Small*, *Zero*, *Positive Small* and *Positive Big*; while the output variable ΔV_{ref} has seven fuzzy membership functions associated with it, namely *Negative Big*, *Negative Medium*, *Negative Small*, *Zero*, *Positive Small*, *Positive Medium* and *Positive Big*. The rule base implemented for the neuro-fuzzy controller is shown in Table 8.2.

Table 8.2. Neuro-fuzzy controller rule base.

Input Fuzzy Variables		$\Delta\omega_4$				
		NB	**NS**	**Z**	**PS**	**PB**
$\Delta\omega_3$	**NB**	PB	PM	PM	PS	Z
	NS	PM	PM	PS	Z	NS
	Z	PM	PS	Z	NS	NM
	PS	PS	Z	NS	NM	NM
	PB	Z	NS	NM	NM	NB

Gaussian membership functions are used for each fuzzy input variable. The membership degree of variable x in the fuzzy set A_j can be expressed as:

$$\mu_{A_j}(x) = \exp\left[-\frac{1}{2}\left(\frac{x - m_j}{\sigma_j}\right)^2\right], \tag{8-4}$$

where m_j and σ_j are the corresponding center and the width of the fuzzy set respectively. Singleton fuzzy sets are assigned to the fuzzy output variable, where the membership degree at a certain singleton point z_j is unity, and zero otherwise:

$$\mu_{B_j}(z) = \begin{cases} 1 & z = z_j \\ 0 & otherwise \end{cases}. \tag{8-5}$$

The parameters of the fuzzy membership functions for the controller input and output variables are summarized in Table 8.3. Parameters m and σ represent the centers and the widths of the fuzzy membership functions for the input variables respectively, while in the last column, parameters z denote the values of the singleton points of the fuzzy membership functions for the output variable.

Table 8.3. Parameters of the fuzzy membership functions for the input/output variables.

Fuzzy Input/Output Variables	$\Delta\omega_3$		$\Delta\omega_4$		ΔV_{ref}
	m	σ	m	σ	z_j
NB	-2.0	1.5	-2.0	1.5	-0.08
NM	---	---	---	---	-0.05
NS	-0.5	0.4	-0.5	0.4	-0.025
Z	0.0	0.4	0.0	0.4	0.0
PS	0.5	0.4	0.5	0.4	0.025
PM	---	---	---	---	0.05
PB	2.0	1.5	2.0	1.5	0.08

The Mamdani min-max method is employed for the fuzzy inference mechanism (Jang and Sun, 1995) and the centroid method (Lee, 1990) is applied for defuzzification of the fuzzy output variable (see Chapter 5 for details). Essentially, the crisp output of the controller can be expressed as Equation (8-6):

$$\Delta V_{ref}(t) = \frac{\sum_j w_j . z_j}{\sum_j z_j}, \tag{8-6}$$

where w_j is the firing strength of the j^{th} rule.

8.3.2. Critic network

The parameters of the neuro-fuzzy controller are derived in such a way that it performs well in a range of operating conditions and during different faults. This is also partly ensured by the inherent robustness of the fuzzy controller. However, the performance is yet far from optimal (see Chapter 6). An adaptive critic designs based approach is therefore applied in order to provide appropriate training signals for the

neuro-fuzzy controller so that it can perform optimal control of the plant during the infinite horizon of the problem. A Critic network is trained in order to learn the cost-to-go function associated with the power system. Once sufficiently trained, the network can provide the training signal for the fuzzy controller.

The utility function is comprised of two terms (decomposed utility function):

$$U(t) = U_1(t) + U_2(t), \tag{8-7}$$

where:

$$U_1(t) = |\Delta\omega_3(t) + \Delta\omega_3(t-1) + \Delta\omega_3(t-2)|, \tag{8-8}$$

$$U_2(t) = |\Delta\omega_4(t) + \Delta\omega_4(t-1) + \Delta\omega_4(t-2)|. \tag{8-9}$$

The two terms are necessary because the rotors of generators 3 and 4 have different swings and therefore, the STATCOM should try to improve the performance of both generators at the same time. The cost-to-go function estimated by the Critic network is:

$$J(t) = \sum_{i=0}^{\infty} \gamma^i . U(t+i). \tag{8-10}$$

This can be further simplified as:

$$J(t) = \sum_{i=0}^{\infty} (\sum_{j=1}^{2} \gamma^i . U_j(t+i)) = \sum_{j=1}^{2} J_j(t). \tag{8-11}$$

Two sub-Critic networks are used, where each one learns one part of the cost-to-go function. Utility function decomposition speeds up the process of Critic network learning, since each sub-Critic is estimating a simpler function (Lendaris and Neidhoefer, 2004).

Figure 8.4 shows the schematic diagram of the Critic network. It consists of two separate MLP neural networks, with 10 neurons in the hidden layer of each one and the same input from the Action network. The hyperbolic tangent is used as the activation function of the hidden neurons.

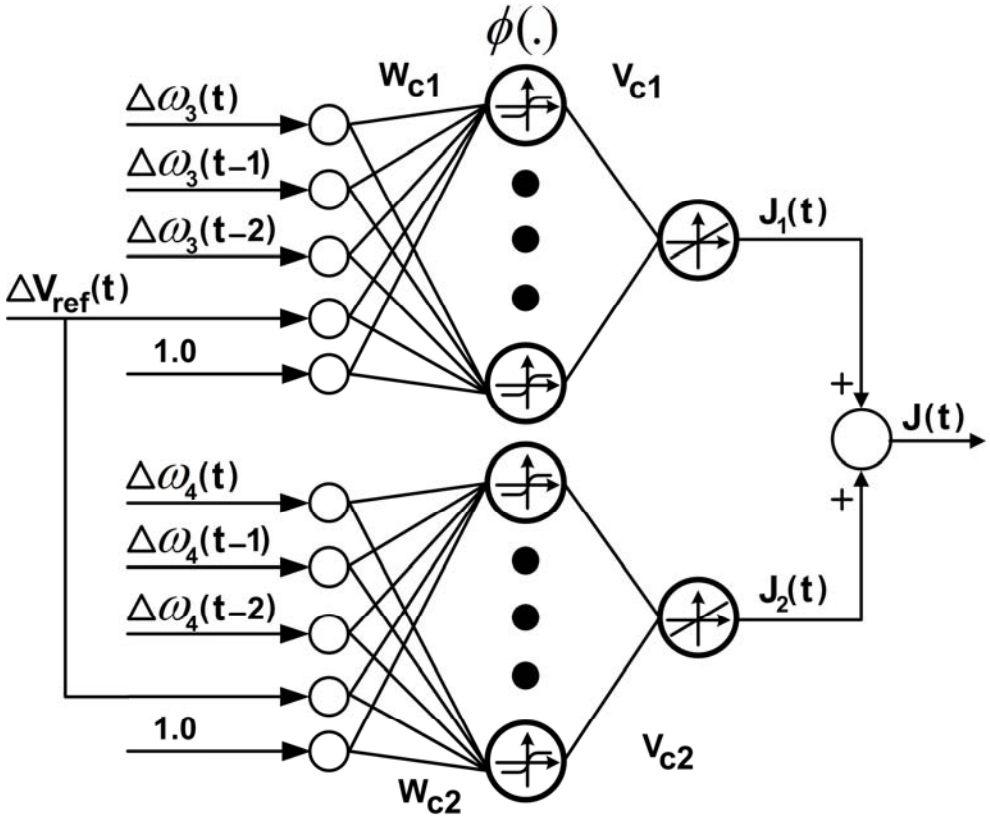

Figure 8.4. Schematic diagram of the STATCOM neuro-fuzzy Critic network.

It has been assumed in this study that the values of the selected system states ($\Delta\omega_3$ and $\Delta\omega_4$) at time (t) are available for the neuro-fuzzy controller. However, in a general power system, there might be transport lags associated with transmitting signals to the controller. Therefore, a wide area state predictor (WASP) system such as the one introduced in Chapter 7 is necessary to predict the correct values of the states at time (t).

8.4. Neuro-Fuzzy External Controller Training

8.4.1. Critic Network Training

A period of forced training is executed for training the neuro-fuzzy Critic network. During this stage, PRBS disturbances are added to voltage reference V_{ref} from an external source (switches S_1 and S_2 in Figure 8.5 are in position 1).

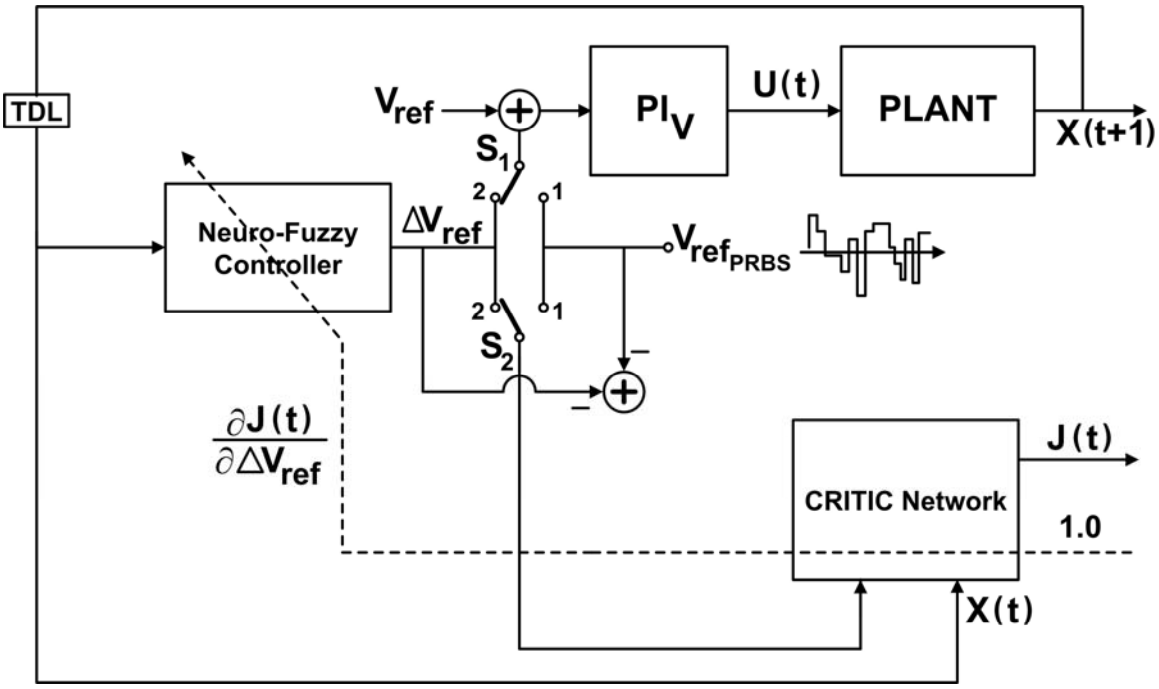

Figure 8.5. STATCOM neuro-fuzzy controller training.

The frequency of the PRBS disturbance is a combination of 0.5, 1 and 2 Hz, with its magnitude limited to $\pm 5\%$. The controller tries to force the plant to follow the reference by injecting the required amount of the reactive power into the network. The resultant deviations in the values of the power system states (Equation (8-3)) are now fed into the Critic network, which goes through backpropagation training to update its synaptic

weight matrices. The training procedure for the Critic network is similar to the process discussed in Chapter 6.

The Critic network training starts with a low discount factor of 0.2, which is gradually increased to 0.8 as the training proceeds. This will help the weights of the Critic network converge faster (see Chapter 6). Moreover, an annealing learning rate scheme is used in which the training starts with a learning rate of about 0.1, which gradually decreases to a value of 0.005. This ensures that during the initial training stages the Critic network adapts itself to the plant dynamics quickly, but as the learning process continues, it does not have drastic reactions to sudden changes in the plant dynamics. In this way the Critic network does not forget the previously learned information.

Critic network training is repeated at various operating conditions until a reasonable accuracy is achieved. Figures 8.6-8.8 show some typical results from the training process of the Critic network.

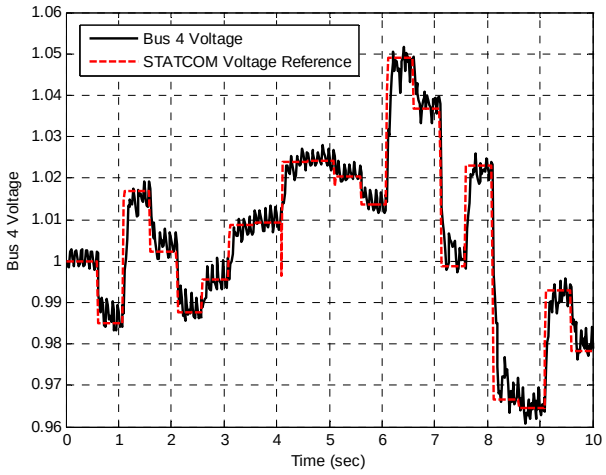

Figure 8.6. STATCOM line voltage reference and the voltage at bus 4 (Figure 8.1) during forced training.

The results show that during the Critic network training with PRBS disturbances, the deviations in the voltage at bus 4 in Figure 8.1 causes oscillations in the rotor speeds of generators 3 and 4 (Figure 8.7). These deviations generate a non-zero utility function and therefore, a positive value for the cost-to-go function *J*. The Critic network in Figure 8.4 is now trained to track the cost-to-go function.

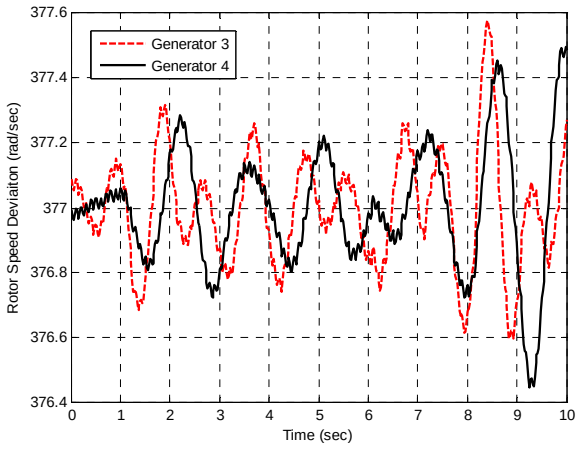

Figure 8.7. Rotor speed deviations of generators 3 and 4 during forced training.

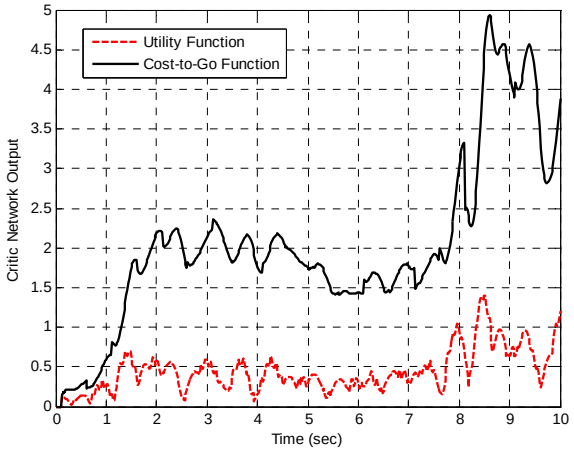

Figure 8.8. Utility function and the estimated values of the cost-to-go function during forced training.

8.4.2. Neuro-Fuzzy Controller Training

The ACD based neuro-fuzzy controller optimizes the overall cost over the time horizon of the problem (minimizing the function J in Equation (8-10)) by providing an optimal control input to the plant. In order for the controller to be able to minimize the cost-to-go function over the infinite horizon of the problem, it should be trained with the following error signal:

$$e(t) = J^*(t) - J(t), \tag{8-12}$$

where $J^*(t)$ is the desired value for the cost-to-go function, which in the case of dealing with deviation signals is zero. The mean-squared error function in Equation (8-13) is used as the objective function for executing the backpropagation algorithm:

$$E(t) = \frac{1}{2} \times e^2(t). \tag{8-13}$$

The coefficients of the fuzzy output variable, i.e., the singleton parameters z_j are considered to be the adaptive parameters of the fuzzy controller. A gradient descent learning algorithm is applied for adjusting these values, where each parameter is updated in the negative direction of the gradient of the objective function $E(t)$ as follows:

$$z_j(t+1) = z_j(t) - \eta \frac{\partial E(t)}{\partial z_j(t)}, \tag{8-14}$$

where η is the learning rate which is considered to be 0.005 in this study. The partial derivative of the objective function with respect to any parameter can be derived using the following chain rule:

$$\frac{\partial E(t)}{\partial z_j(t)} = \frac{\partial E(t)}{\partial J(t)} \times \frac{\partial J(t)}{\partial \Delta V_{ref}(t)} \times \frac{\partial \Delta V_{ref}(t)}{\partial z_j(t)}. \tag{8-15}$$

The first term in Equation (8-15) is equal to $J(t)$ and the second term can be derived by backpropagating constant 1.0 through the neuroidentifier (see Appendix D). The reader is referred to (Feldman, 1996) for more mathematical details. The last term in Equation (8-15) can also be simplified as follows:

$$\frac{\partial \Delta V_{ref}(t)}{\partial z_j(t)} = \frac{w_j \times \Delta V_{ref}(t)}{\sum_j w_j}. \tag{8-16}$$

In this study only the polynomial function coefficients of the neuro-fuzzy controller are updated. It is also possible to apply a *full updating* scheme where the parameters of the input membership functions in Equation (8-4) are adaptively adjusted as well. However, a *partial updating* scheme is used here which considers all the parameters of the input fuzzy membership functions to be constant throughout the simulation. This approach is adopted since it is less computationally intensive. Moreover, the parameters of the input fuzzy sets are derived after closely studying the performance of the two generators during various natural faults applied to the system.

The ACD neuro-fuzzy controller is trained by the cost function defined in Equation (8-10) using the update formula in Equations (8-15)-(8-16), so that its output coefficients are adjusted for optimum performance. The fuzzy controller is trained online during the actual performance of the power system. At this stage the fuzzy controller is controlling the plant (switches S_1 and S_2 in Figure 8.5 are in position 2). Various faults and disturbances are applied to the power network and the resultant error signal derived in Equation (8-15) is used for updating the parameters z_j of the output fuzzy sets.

Figures 8.9 and 8.10 illustrate some typical simulation results during the training stage of the neuro-fuzzy controller. The results are related to the case when a 100 *ms*

three phase short circuit is applied to the transmission line connecting buses 2 and 5 after 1 sec. It can be seen that as the fault occurs, the cost-to-go function $J(t)$ is drastically increased and the neuro-fuzzy controller responds to the fault according to its rule base.

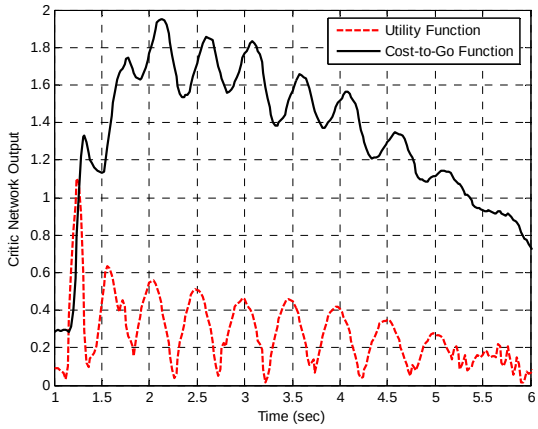

Figure 8.9. Utility function and the cost-to-go function during controller training.

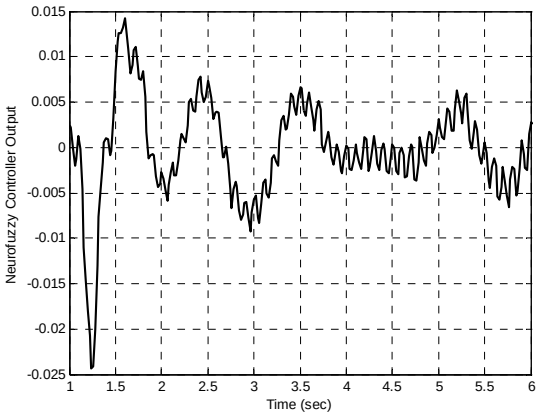

Figure 8.10. Output of the neuro-fuzzy external controller.

Figure 8.11 illustrates the corresponding training signal that is applied for updating the parameters of the fuzzy controller. An adaptive learning rate parameter is used for the

fuzzy controller that is set to 0.01 during the normal operation of the controller and is reduced to 0.0005 if the magnitude of the training signal exceeds 0.05. This is to avoid instability of the fuzzy controller parameters as a result of large error spikes.

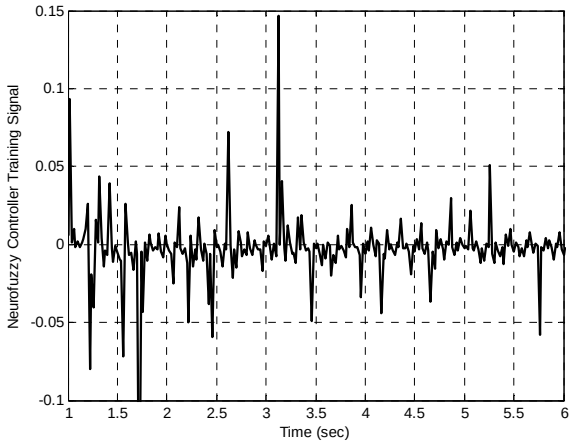

Figure 8.11. Error signal used for adjusting the fuzzy controller parameters.

8.5. Simulation Results

8.5.1. Case Study 1: Three Phase Short Circuit at the Transmission Line 2-5

In the first case study, a 100 *ms* three phase short circuit is applied at the middle of the transmission line connecting buses 2 and 5. The line is reconnected after the fault is cleared. Figures 8.12 and 8.13 show some typical simulation results. It can be seen that the proposed neuro-fuzzy controller manages to improve the dynamic damping of both generators, even though the rotors of the two machines have different, and opposing at times, excursions.

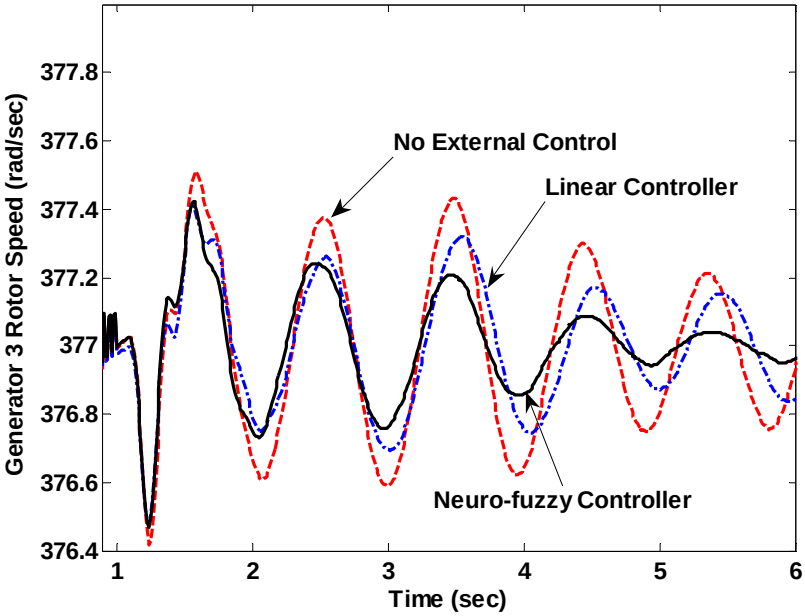

Figure 8.12. Generator 3 speed deviations during case study 1.

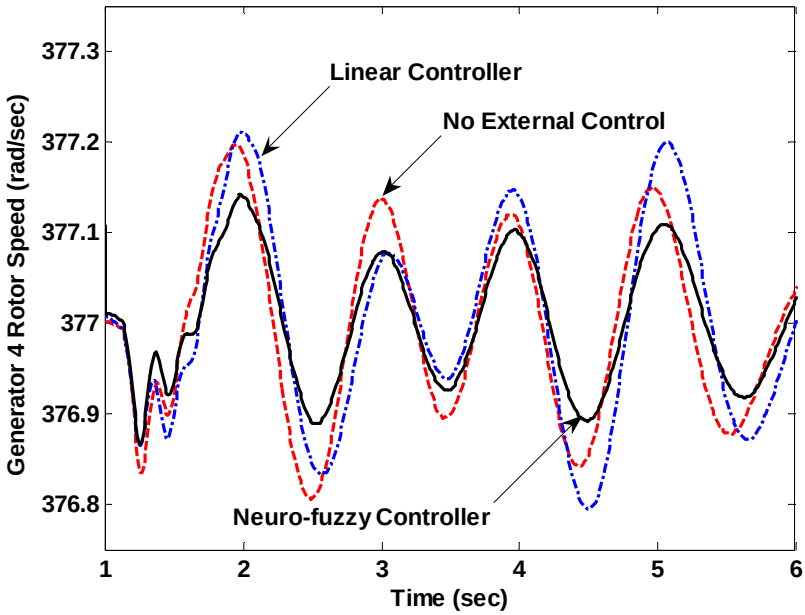

Figure 8.13. Generator 4 speed deviations during case study 1.

Figure 8.14 illustrates the reactive power injected by the STATCOM during the fault. It can be seen that the neuro-fuzzy controller prevents the STATCOM from injecting a large amount of power into the network at the beginning of the fault. It reduces the reactive power injection by 15 and 5 MVar compared to the case of having no external controller and the linear external controller respectively. This will help reduce the currents passing through the STATCOM inverter switches. However, it makes the STATCOM inject reactive power afterwards in an intelligent way to help damp out the oscillations.

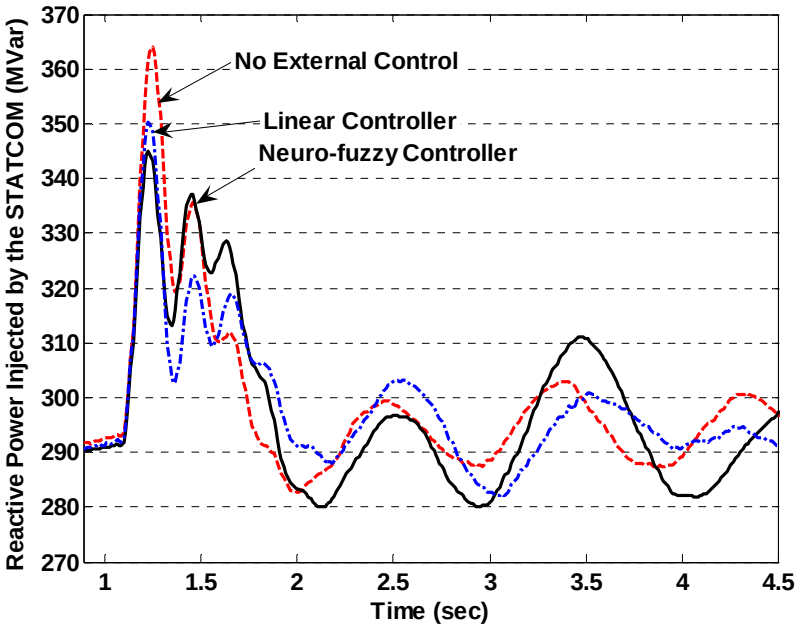

Figure 8.14. Reactive power injected by the STATCOM during case study 1.

8.5.2. Case Study 2: Three Phase Short Circuit at the Transmission Line 3-4

A 100 *ms* three phase short circuit is now applied to the transmission line connecting the STATCOM to generator 3. Figures 8.15 and 8.16 show the effectiveness of the proposed neuro-fuzzy controller in damping out the rotor speed oscillations.

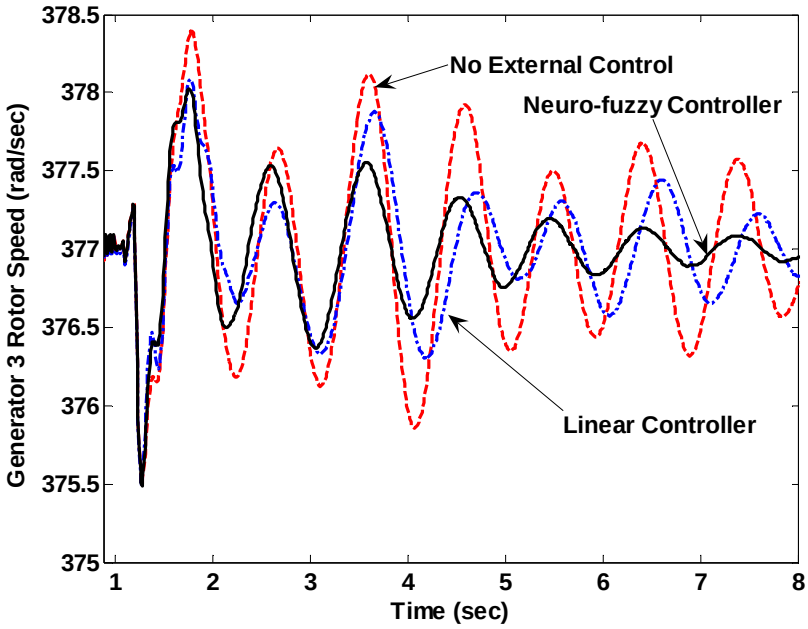

Figure 8.15. Generator 3 speed deviations during case study 2.

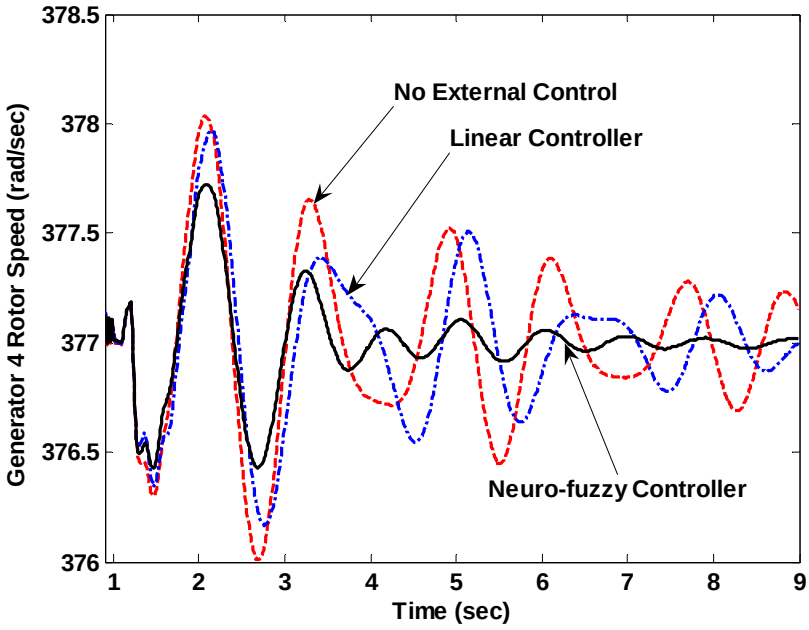

Figure 8.16. Generator 4 speed deviations during case study 2.

Clearly, improving the dynamic damping of the synchronous generators helps damp out the low frequency oscillations in the transmission line active power flow. Figure 8.17 shows the positive impact of the proposed neuro-fuzzy controller on the active power flow of the transmission line connecting the STATCOM to generator 4.

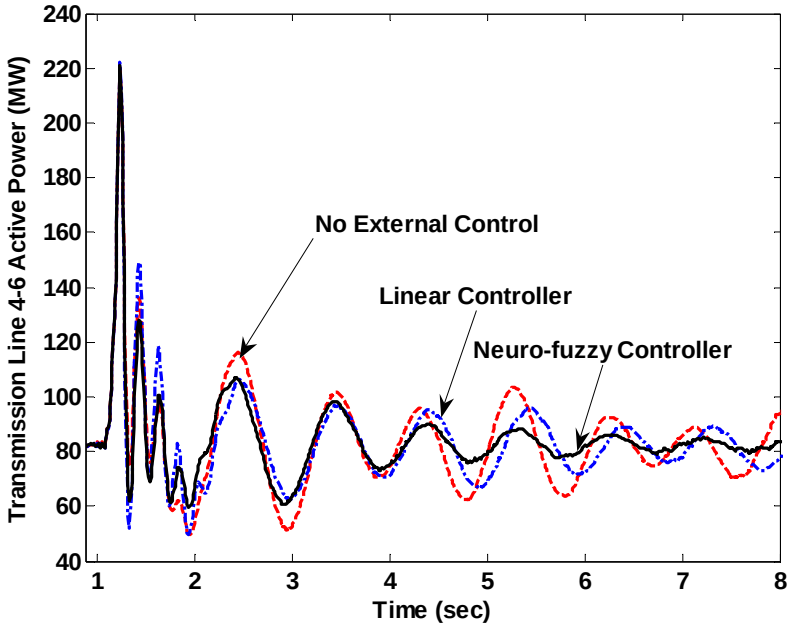

Figure 8.17. Active power flow through the transmission line 4-6 during case study 2.

8.5.3. Case Study 3: Three Phase Short Circuit at the Transmission Line 7-8

A 100 *ms* three phase short circuit is applied to the transmission line connecting buses 7 and 8. This section of the power system is relatively weak and sensitive to disturbances. Generator 3 is also close to the line and is therefore largely affected by any fault in this area. Figures 8.18 and 8.19 illustrate the effectiveness of the neuro-fuzzy external controller in restoring the system back to the steady state condition.

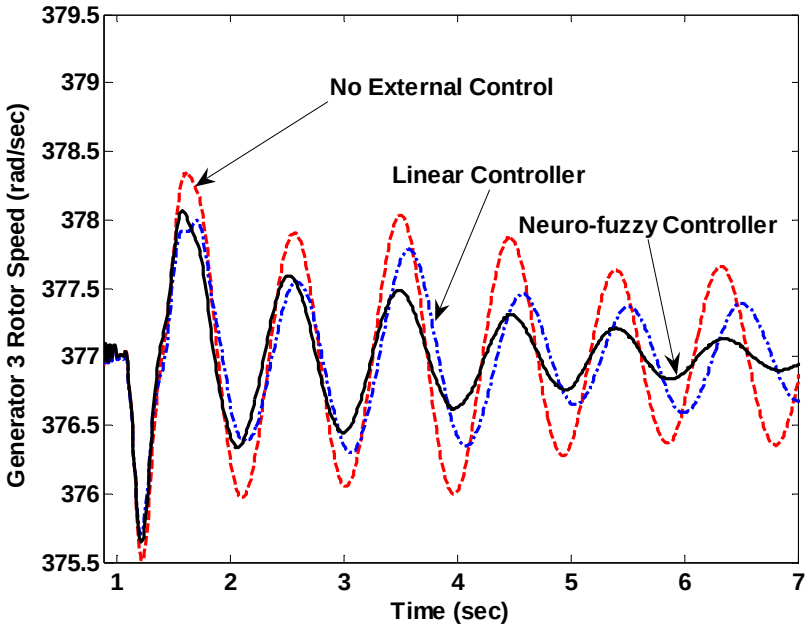

Figure 8.18. Generator 3 speed deviations during case study 3.

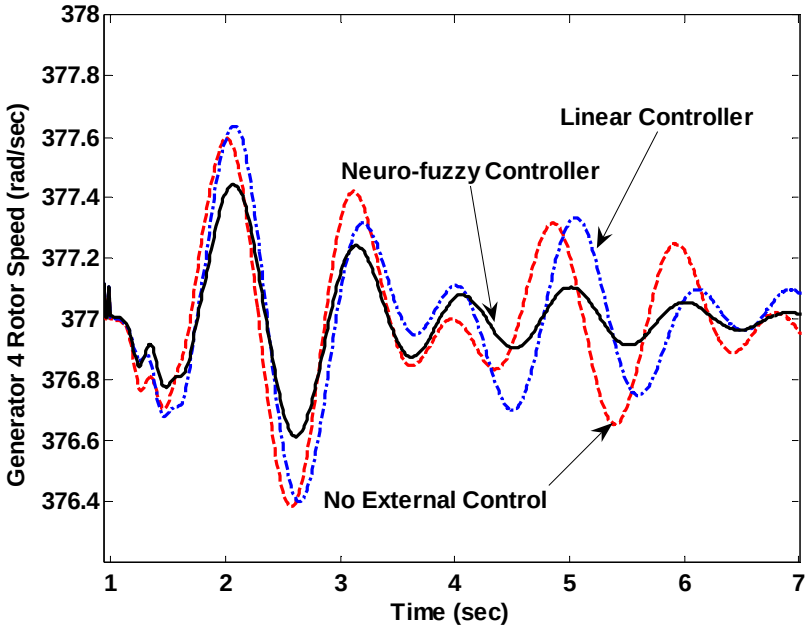

Figure 8.19. Generator 4 speed deviations during case study 3.

Figure 8.19 shows that the linear controller, which was trained at a different operating condition, cannot respond effectively to the rotor speed deviations of generator 4. Similar results are shown in Figure 8.20 to illustrate that the linear controller fails to quickly restore the steady state conditions of the system, whereas the neuro-fuzzy controller effectively damps out the active power oscillations.

Figure 8.21 emphasizes on the fact that the STATCOM externally controlled by the neuro-fuzzy controller injects less initial reactive power into the network for responding to the fault.

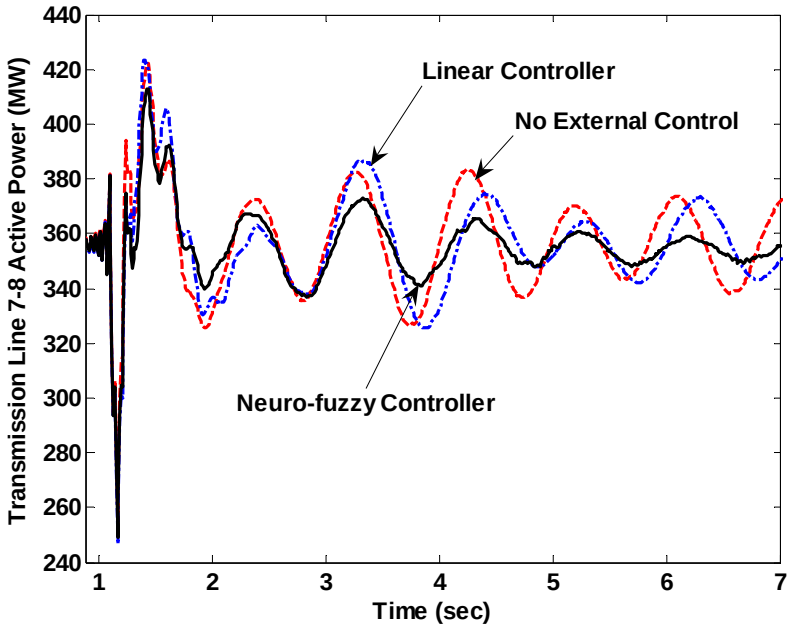

Figure 8.20. Active power flow through the transmission line 7-8 during case study 3.

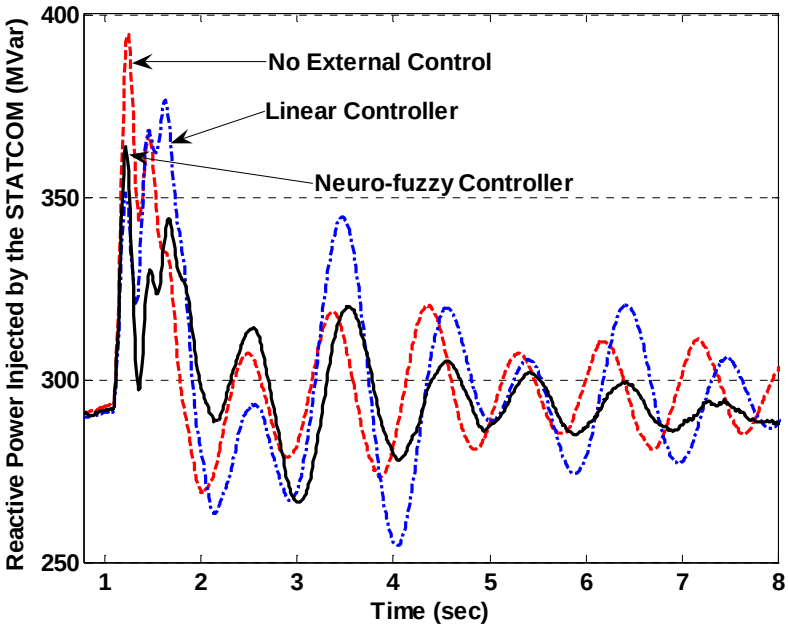

Figure 8.21. Reactive power injected by the STATCOM during case study 3.

8.5.4. Case Study 4: Transmission Line 2-5 Switch On/Off

In a different type of disturbance, the transmission line connecting buses 2 and 5 is disconnected and is switched back into the power system after 3 seconds. Figures 8.22 and 8.23 show the superiority of the proposed neuro-fuzzy external controller in damping out the rotor speed deviations fast.

The effect of the neuro-fuzzy external controller can also be evaluated by comparing the reactive power injection from the STATCOM during the transient. Figure 8.24 shows the poor performance of the linear external controller, which commands almost 500 MVar from the inverter during the first two seconds following the fault. This occurs because the gains of the linear controller are not optimally tuned, rather they are derived based on a trial and error approach.

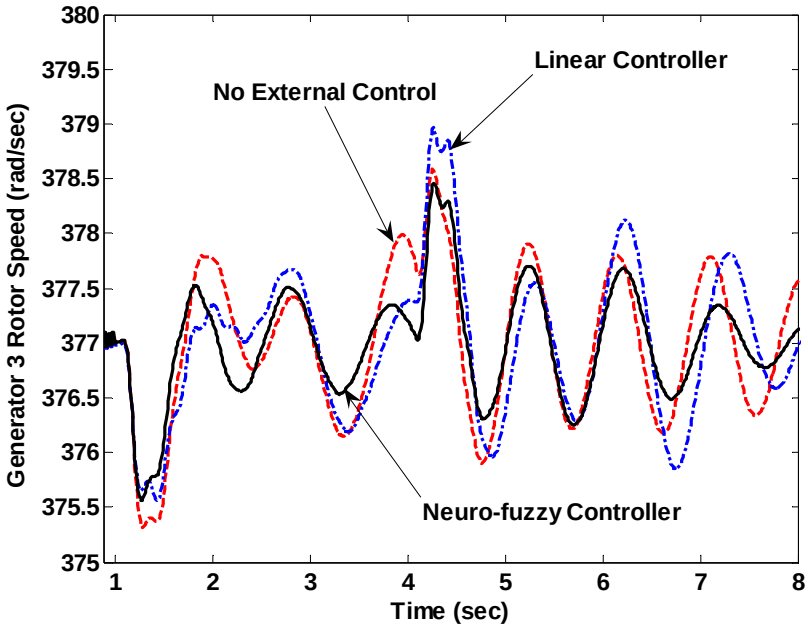

Figure 8.22. Generator 3 speed deviations during case study 4.

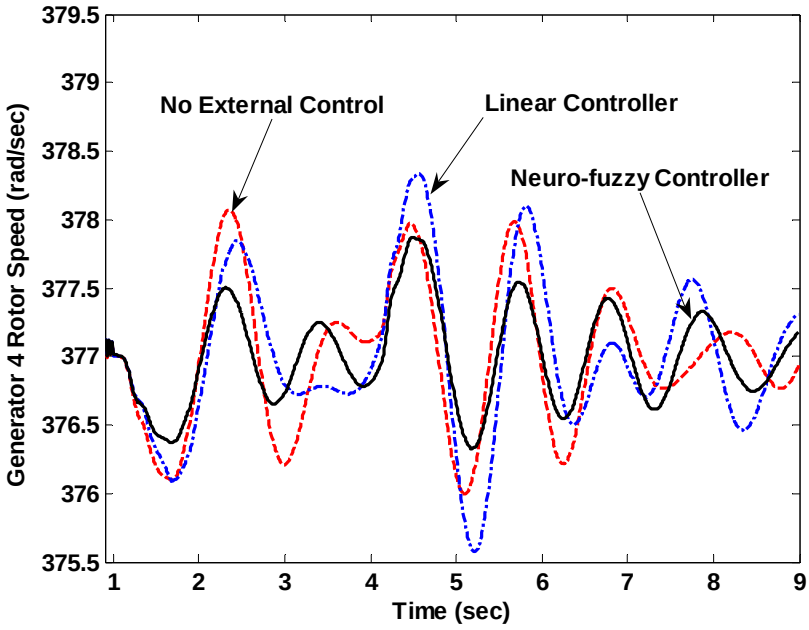

Figure 8.23. Generator 4 speed deviations during case study 4.

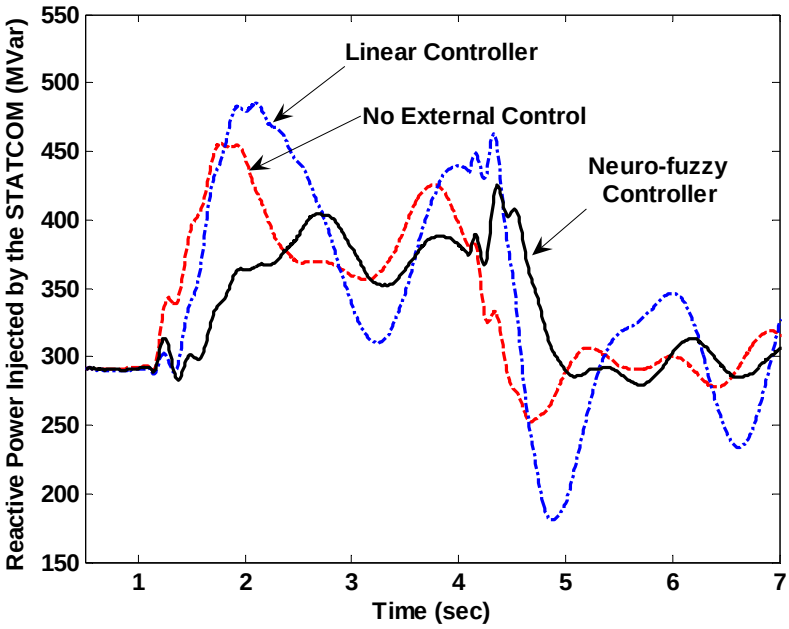

Figure 8.24. Reactive power injected by the STATCOM during case study 4.

8.5.5. Case Study 5: Transmission Line 4-6 Outage

So far, it was assumed in all the cases studied that none of the lines that connect the STATCOM to the synchronous generators are disconnected. However, it can be seen in Figure 8.1 that any fault that causes outage of the transmission lines 3-4 or 4-6 temporarily or permanently, will make the STATCOM to lose its impact on the corresponding generator.

In this section, a 100 *ms* three phase short circuit is applied at 1 sec to the transmission line connecting the STATCOM to generator 4, and the line is disconnected as a result of the fault. Clearly, due to the loss of a direct path, the STATCOM cannot have a drastic impact on the rotor speed oscillations of generator 4. Therefore, the rule base in Table 8.2 cannot be applied in its present form in order to derive the fuzzy controller output. One solution would be deriving a backup rule base for the fuzzy

controller in case one of the power system states is not directly affected by the STATCOM (due to some configuration change). However, an easier and more practical way to deal with temporary line/bus outages that disconnect the STATCOM from one of the generators would be removing that generator's speed deviation from the rule base by considering its value is zero. In this way the output of the fuzzy controller is generated based on the rotor speed deviations of one generator only.

Figure 8.25 depicts the speed deviations of generator 3. It can be seen that the STATCOM with the proposed neuro-fuzzy controller can effectively damp out the oscillations. With the above assumption applied to the rule base, a fuzzy controller with fixed parameters would under-compensate the speed deviations of generator 3 due to the way the rule base is defined. However, the proposed neuro-fuzzy controller does not suffer from this due to the continuous adjustment of the fuzzy controller output parameters by the Critic network.

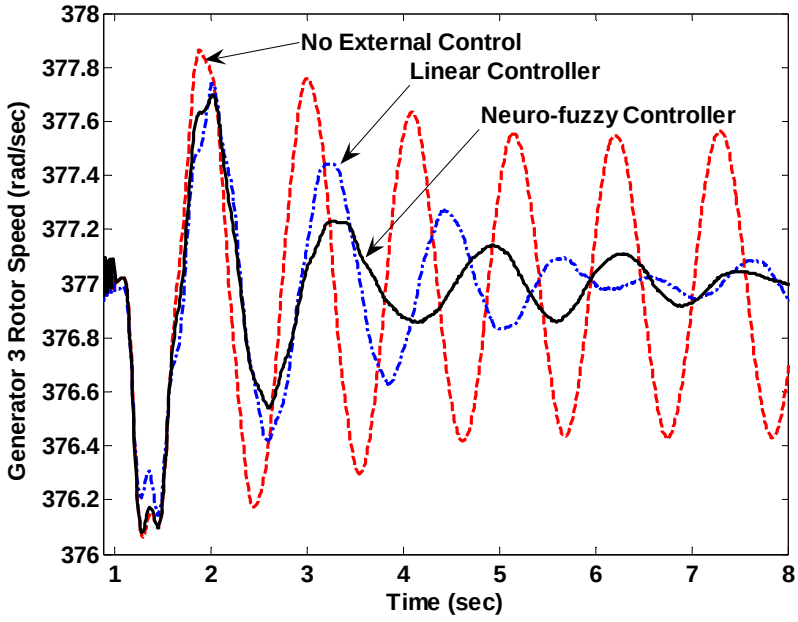

Figure 8.25. Generator 3 speed deviations during case study 5.

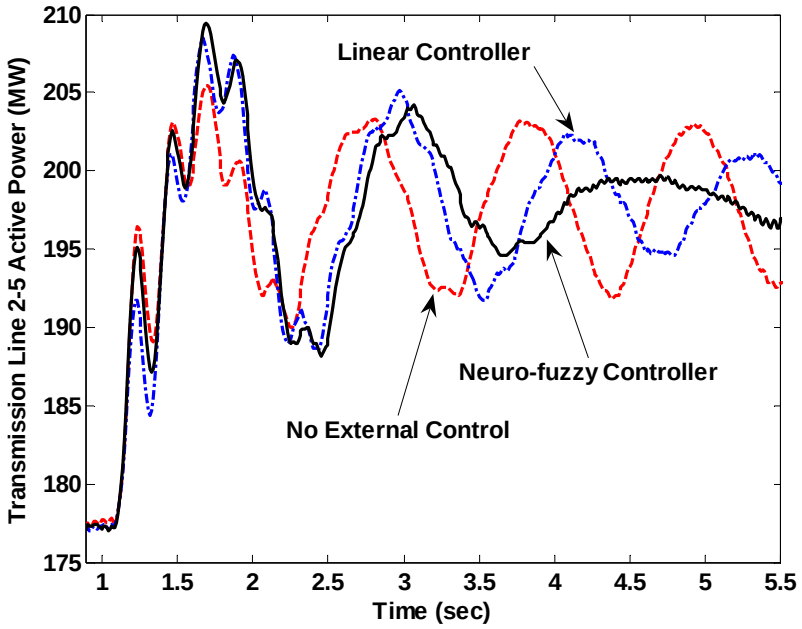

Figure 8.26. Active power flow through the transmission line 2-5 during case study 5.

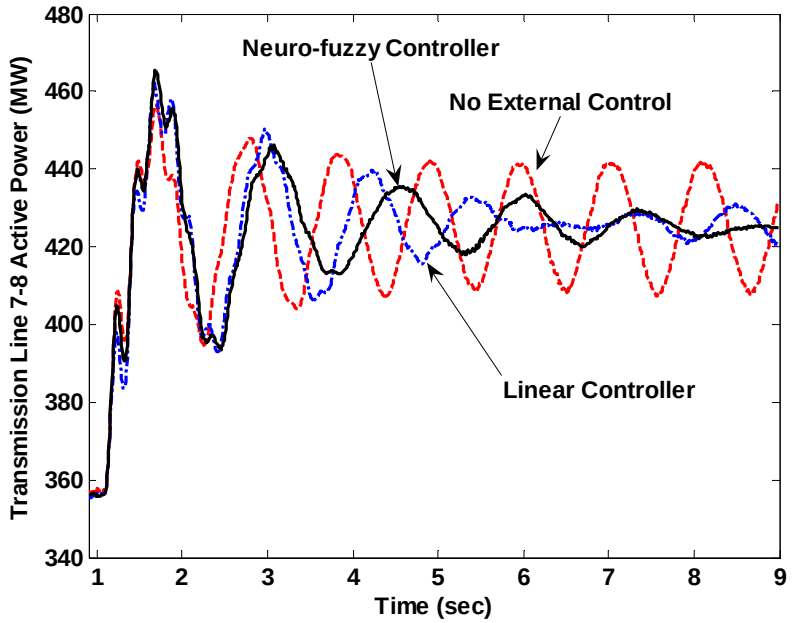

Figure 8.27. Active power flow through the transmission line 7-8 during case study 5.

Figures 8.26 and 8.27 once again show that the neuro-fuzzy external controller can damp out the active power low frequency oscillations in the transmission lines 3-4 and 7-8 far more effectively than even the linear external controller. In all these simulation results, the linear external controller has also been reduced to linear scheme 1 in Figure 8.2.

8.5.6. Performance Measurement

In this section, the performance of the neuro-fuzzy external controller is compared with the linear external controller and the uncompensated power system. A performance index is defined for each case as in Equation (8-1). Table 8.4 summarizes the results. The overall performance index of each control scheme in Table 8.4 is defined as in Equation (8-2). In the last row of the Table 8.4 the overall performance indices are normalized based on the overall performance index of the neuro-fuzzy external controller.

Table 8.4. Performance indices of the neuro-fuzzy and the linear external controllers for the STATCOM.

Type of Test	Performance Index		
	No External Controller	Linear External Controller	Neuro-fuzzy External Controller
Case Study 1	1.98	2.32	2.64
Case Study 2	0.98	1.25	1.68
Case Study 3	1.24	1.57	2.12
Case Study 4	0.57	0.53	0.67
Case Study 5	1.26	1.58	1.76
Overall Performance Index	0.204	0.227	0.285
Normalized Overall Performance Index	0.71	0.79	1

8.6. Conclusions

An intelligent external controller is designed in this chapter for a STATCOM connected to the IEEE 12-bus benchmark system. The controller provides auxiliary reference signal for the STATCOM line voltage control loop and enables the FACTS device to improve the dynamic stability of the power system during large scale faults and disturbances. A neuro-fuzzy approach is employed in this study which allows for the representation of the control actions in terms of heuristic terms and reasoning. The performance of the controller is further improved by incorporating an adaptive critic designs based neural network technique for providing optimal control over the infinite horizon of the problem.

The complexity and the nonlinearity of the system do not allow a conventional linear technique to effectively control the plant. However, simulation results indicate that the proposed neuro-fuzzy controller is capable of significantly improving the dynamic stability of the power system during various faults at different operating conditions.

CHAPTER 9

ADAPTIVE CRITIC DESIGNS BASED WIDE AREA NEUROCONTROLLER FOR THE IEEE 12-BUS BENCHMARK SYSTEM

9.1. Introduction

In the previous chapter, it was shown that by employing global (wide area) signals, a STATCOM can contribute to the system damping during small scale as well as large scale disturbances. This chapter extends the previous idea by implementing a wide area neural network based optimal controller for the IEEE 12-bus benchmark system. It is also assumed that a STATCOM is connected to bus 4 in this power system in order to provide extra voltage support and improve the dynamic stability. The proposed wide area controller (WAC) provides auxiliary reference signals for the automatic voltage regulators (AVR) of the three generators, as well as the line voltage control loop of the STATCOM.

An action dependent adaptive critic designs (ADACD) based technique, which is a model-free design, is used for implementing the WAC. A neural network based Critic network is trained to learn the cost-to-go function associated with the control objective of the power system. Once converged, this network can estimate the cost-to-go function at every operating point over the infinite horizon of the problem. Training signals provided by the Critic network enable the Action network to optimally control the plant, without being affected by its size, nonlinearity and uncertainties associated with it.

A very important aspect of such a WAC scheme is to ensure that the auxiliary signals are provided for the local controllers in such a way that unwanted interactions between

them are avoided. This is achieved by using a functional link (FNL) neural network structure for implementing the Action network. Detailed explanations are provided in the following sections.

It is also assumed in this chapter that access to the required plant information in real-time is provided by using a robust wide area state predictor (WASP), as explained in Chapter 7.

9.2. Power System Stabilizer (PSS)

Excitation systems with high gains and fast response time can significantly improve the transient stability of a power system by opposing the voltage drop at the generator busses. However, these can create negative damping for the low frequency electromechanical oscillations in the system, causing dynamic instability (Kundur, 1994). In order to solve this problem, large gains are normally used in the excitation system for achieving appropriate voltage regulation and improving transient stability; however, a *Power System Stabilizer* (PSS) is employed in order to provide more dynamic stability for the generator (Kundur, 1994). Figure 9.1 illiterates the schematic diagram of a typical PSS.

The PSS generates a torque component $\Delta T_{e,PSS}$ that ideally should be in phase with the rotor speed deviation $\Delta\omega$ in order to provide dynamic damping; therefore, the ratio of $\Delta T_{e,PSS} / \Delta\omega$ should be a positive real number. This term can be expanded as:

$$\frac{\Delta T_{e,PSS}}{\Delta\omega} = \frac{\Delta T_e}{V_{PSS}} \times \frac{V_{PSS}}{\Delta\omega} . \tag{9-1}$$

The term $V_{PSS} / \Delta\omega$ in Equation (9-1) is the transfer function of the PSS. It can be shown that in a synchronous generator the term $\Delta T_e / V_{PSS}$ has a lagging phase; therefore,

the transfer function of the PSS should have a leading phase so that it can provide positive damping for the oscillatory modes (Kundur, 1994).

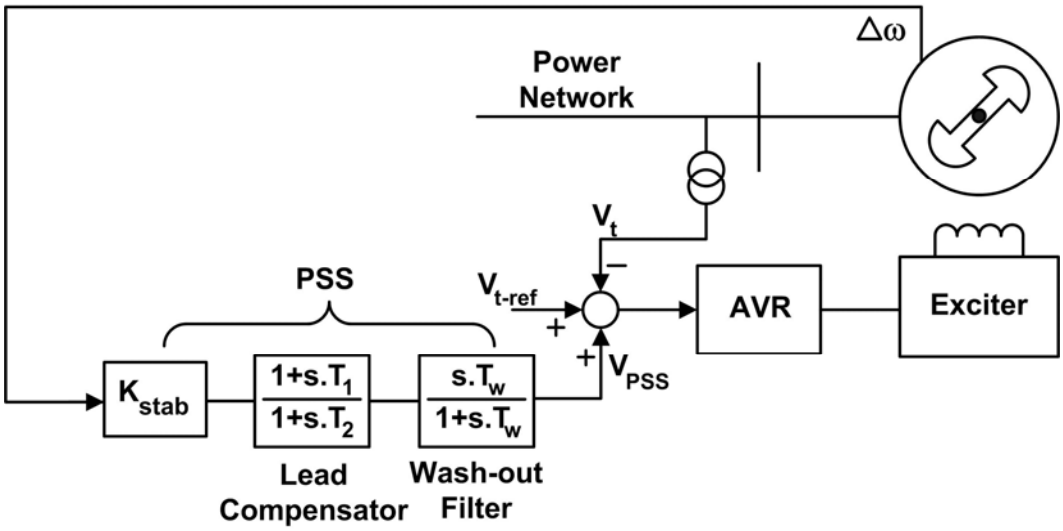

Figure 9.1. Schematic diagram of a power system stabilizer (PSS).

Ideally, the PSS is designed in a way that it provides 100% compensation, so that the torque generated by the PSS is completely in phase with the rotor speed deviations ($\Delta T_{e,PSS} / \Delta\omega$ is a positive real number). However, changes to the power system configuration or the operating point can change the parameters of the synchronous generator model, since it is derived by linearizing the power system equations at a specific operating condition. This will in turn change the magnitude and angle of the torque $\Delta T_{e,PSS}$ generated by the PSS (Figure 9.2.a). If this vector is moved to the left side of the $\Delta\delta - \Delta\omega$ plane, it will generate a negative synchronizing torque which will deteriorate the transient stability of the system. Hence, in practice, the degree of phase compensation is designed for less than 100% (Figure 9.2.b) and a safety margin is

considered so that the PSS never generates negative synchronizing torque (Kundur, 1994).

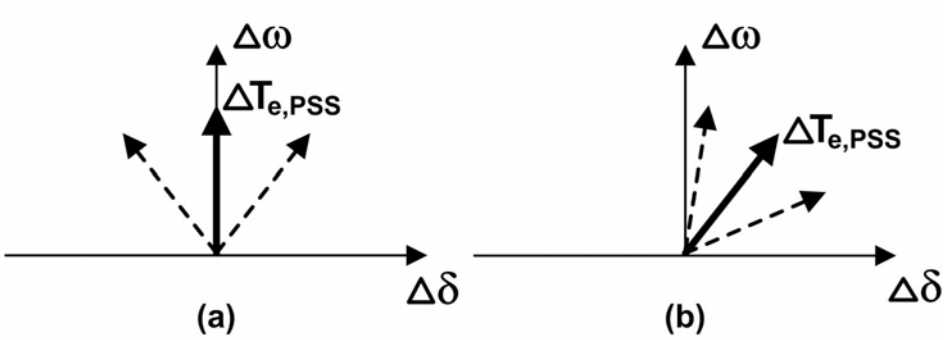

Figure 9.2. Electric torque generated by the PSS:
a. 100% compensation, b. under-compensation.

The PSS constant gain k_{stab} should be large enough to provide damping for the system modes, yet it should be small enough not to cause instability for the other modes of the system, such as the excitation system.

The wash-out filter is used in order to prevent the PSS from responding to slow dynamic changes in the generator speed, since these deviations should be responded to by the speed governor.

9.3. Wide Area Control in a Power System

In a simple power system with a fixed structure, a PSS can be effectively tuned to damp out a specific mode. However, the frequencies of the oscillatory modes are highly dependent on the operating conditions of the system, and it changes as the network configuration, load level and operating conditions change (Kundur, 1994). In such cases, 100% phase compensation in the PSS design might at times lead to negative transient

damping (Figure 9.2). Therefore, the PSS parameters are normally tuned to provide under-compensation. This way, the PSS can provide positive damping at a range of frequencies, even if the operating conditions of the power system change.

Moreover, a locally tuned and designed PSS normally fails to perform effectively during the inter-area oscillations; although a carefully tuned PSS may be effective in damping inter-area modes up to a certain transmission loading. The effectiveness of the local PSS in damping inter-area oscillatory modes is limited because these modes are not as highly controllable and observable in the generator's local signals as the local modes (Klein *et al.*, 1992).

In addition, the number of dominant modes in the system is much larger than the number of the controlled devices available (Chow *et al.*, 2000). Therefore, every local PSS should ideally be able to respond to various oscillatory modes in the power system.

A hierarchical control structure can be a solution to the above problems. Several authors have investigated the issue of designing a PSS using local as well as global signals (Chow *et al.*, 2000, Aboul-Ela *et al.*, 1996). Venayagamoorthy and Ray (2005) have also proposed a neural network based optimal controller as a supervisory level PSS which can effectively target the inter-area oscillation modes. All these techniques use global signals (speed deviations of the remote generators) in order to provide an additional control signal for the local generators.

In this chapter the ideas set forth in the above references are extended by designing a WAC that not only employs global signals for hierarchical control of the generators, but also incorporates the STATCOM controller into the supervisory level control in order to provide more overall damping for the power system.

Figure 9.3 shows the schematic diagram of the IEEE 12-bus system with the WAC. The WAC receives the system states/measurements, analyzes the data and sends the appropriate auxiliary signals to the voltage references of the generators' AVRs and the STATCOM line voltage controller. Since the global measurements throughout the network will reach the WAC with a transport lag, a WASP is used to compensate for the delay. The WASP receives different states/measurement from across the power system and based on the algorithm explained in Chapter 7, it estimates the states of the system in real-time. Since the proposed WAC is designed to improve the dynamic stability during the various disturbances applied to the power system, the WASP is trained in order to estimate the rotor speed deviations for the three generators.

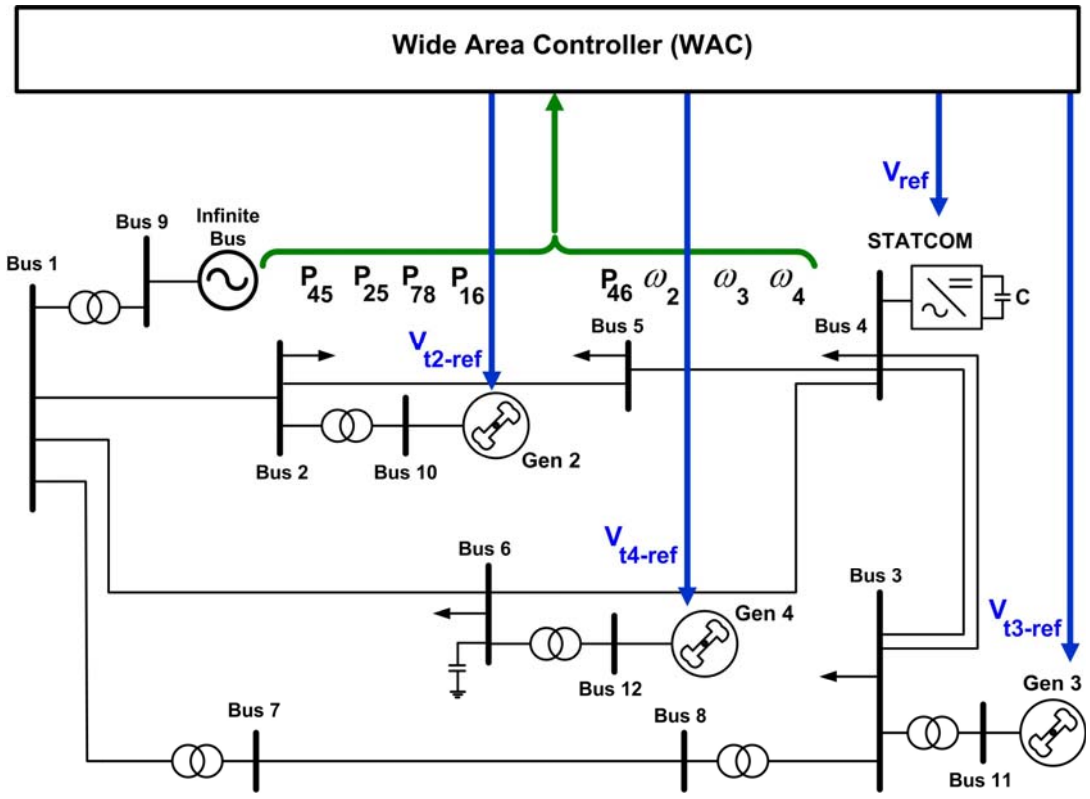

Figure 9.3. Schematic diagram of the IEEE 12-bus benchmark system with wide area controller.

In general, a power system can have three levels of control (Wang, 2001): The primary level is the control actions performed by the local controllers. At the secondary level, the loop of the local controllers' reference commands are closed by means of a supervisory level controller that determines their set-points. The third and highest level of control comprises of a supervisor that uses global information throughout the network to optimize certain characteristics of the power system. The second and third level control can be implemented using adaptive critic designs (ACD) based techniques. An ACD based wide area neurocontroller can serve as a hierarchical controller for the power system. While the primary level still comprises of the local controllers of the components in the power network, the Action network performs as a secondary level controller which closes the loop of the local controllers' set-points. The optimization responsibilities of the third level control are accomplished by the Critic network that tries to optimize the system performance by providing the appropriate training signals for the Action network. Figure 9.4 illustrates the schematic diagram of the ACD based hierarchical controller.

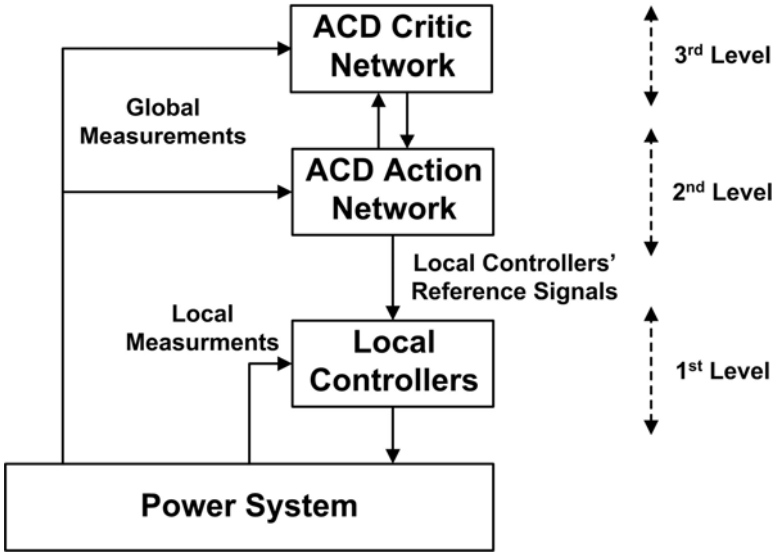

Figure 9.4. Schematic diagram of the ACD based hierarchical controller.

Figure 9.5 shows the schematic diagram of the WAC. It is an ACD based neurocontroller that consists of a Critic network, which is trained to estimate the cost-to-go function $J(t)$, and an Action network that provides the auxiliary reference signals for the three generators' AVRs and the STATCOM voltage reference. The vector $\underline{V}_{ref}$ in Figure 9.5 denotes the vector of steady state voltage set-points for the AVRs of the three generators and the STATCOM line voltage control loop.

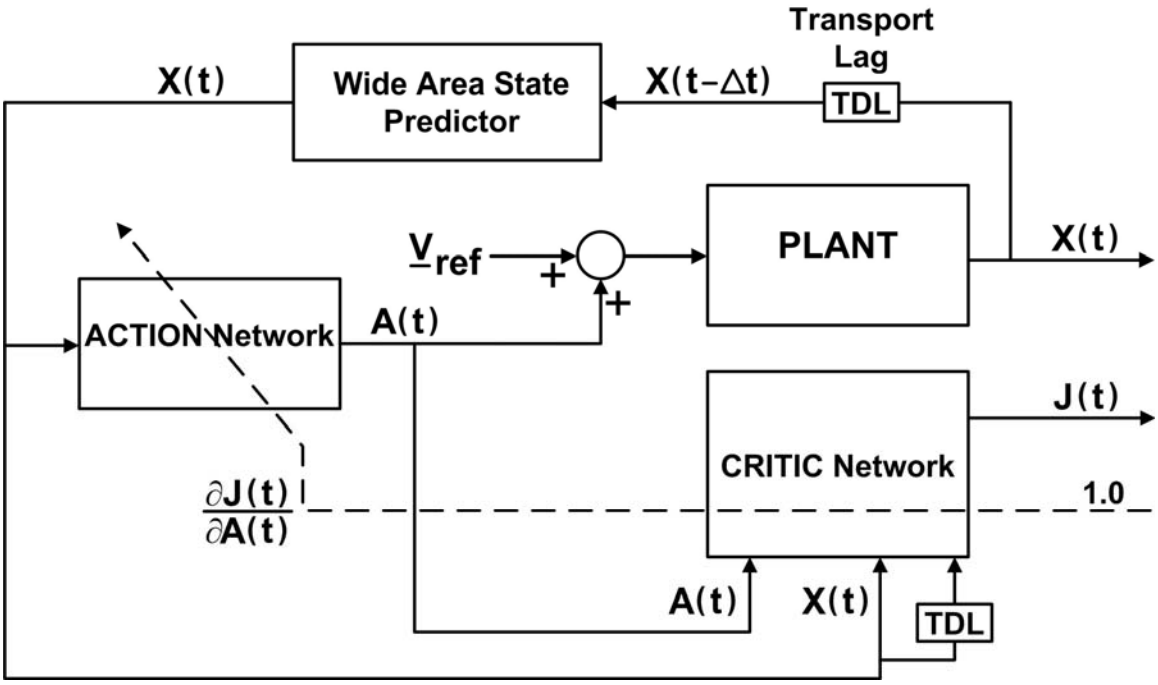

Figure 9.5. Schematic diagram of the WAC.

9.4. WAC Neurocontroller

9.4.1. Utility Function Decomposition

The vector of the states of the power system is considered to be comprised of the speed deviations of the three generators in Equation (9-2):

$$X(t) = [\Delta\omega_2(t), \Delta\omega_3(t), \Delta\omega_4(t)]^T . \quad (9\text{-}2)$$

Similar to the approach presented in Chapter 8 three separate utility function components U_1, U_2 and U_3 are defined for the WAC:

$$U(t) = U_1(t) + U_2(t) + U_3(t) , \quad (9\text{-}3)$$

where:

$$U_1(t) = |\, \Delta\omega_2(t) + \Delta\omega_2(t-1) + \Delta\omega_2(t-2) \,| , \quad (9\text{-}4)$$

$$U_2(t) = |\, \Delta\omega_3(t) + \Delta\omega_3(t-1) + \Delta\omega_3(t-2) \,| , \quad (9\text{-}5)$$

$$U_3(t) = |\, \Delta\omega_4(t) + \Delta\omega_4(t-1) + \Delta\omega_4(t-2) \,| . \quad (9\text{-}6)$$

The utility function decomposition applied here is due to the fact that the rotors of the three generators have different mechanical swings and therefore, the WAC should try to improve the performance of all three at the same time. If they are all considered in one single utility function, they might at times cancel out each other's effect, which can result in, for example, indicating a low utility function during a large fault. The cost-to-go function estimated by the Critic network is:

$$J(t) = \sum_{i=0}^{\infty} \gamma^i . U(t+i) , \quad (9\text{-}7)$$

which can be further simplified as:

$$J(t) = \sum_{i=0}^{\infty} (\sum_{j=1}^{3} \gamma^i . U_j(t+i)) = \sum_{j=1}^{3} J_j(t) . \quad (9\text{-}8)$$

9.4.2. Critic Network Structure

Three sub-Critic networks are used, one for each of the three utility functions U_1, U_2 and U_3 respectively, where each one learns one part of the cost-to-go function. Figure 9.6 shows the schematic diagram of the Critic network. It consists of three separate MLP neural networks, with 10 neurons in the hidden layer of each one and the hyperbolic tangent as the activation function of the hidden neurons. Since an action dependent ACD

controller design is adopted in this study, the controller outputs need to be incorporated into the input vectors of the three sub-Critic networks as well. The ACD neurocontroller in Figure 9.5 generates four auxiliary reference signals for the power system:

$$A(t) = [\Delta V_{t2-ref}(t), \Delta V_{t3-ref}(t), \Delta V_{t4-ref}(t), \Delta V_{ref}(t)]^T . \quad (9\text{-}9)$$

For an action dependent Critic network, it is possible to feed all the four control outputs to each sub-Critic. However, not all the control outputs affect all the state variables in Equation (9-2). In fact, some of the control outputs have negligible effect on certain state variables. For instance, synchronous generators 2 and 3 are far from one another and therefore, the effect of the auxiliary control signal applied to the AVR of each one on the rotor speed deviations of the other one can be ignored. Simplifying the structures of the sub-Critics by removing the unnecessary control outputs can help speed up the training process of each sub-Critic network. According to the discussion above, the structure of the Critic network is illustrated in Figure 9.6.

The three sub-Critics can be trained independently one-by-one, or simultaneously. Nevertheless, the scheme used for training each sub-Critic is similar to the method described for a single Critic network in Section 6.3.1.

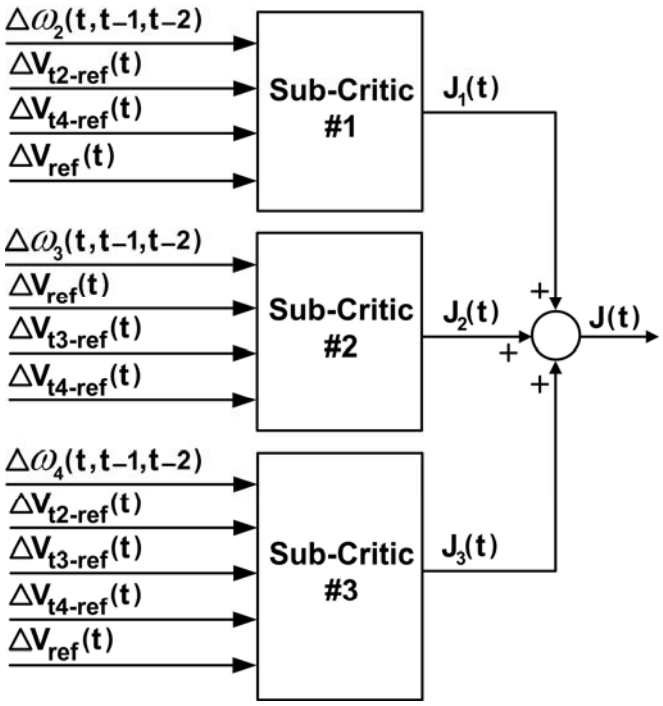

Figure 9.6. Schematic diagram of the WAC Critic network.

9.4.3. Action Network

As mentioned in Chapter 1, one of the main objectives of having a wide area control scheme in the power network is to be able to respond to a system disturbance with the least amount of control effort possible. This means that the WAC needs to ensure that undesired interactions between the controllers are reduced.

The WAC Action network is responsible for generating the desired auxiliary reference signals for the generators' AVRs and the STATCOM voltage loop PI_V controller. Traditionally, a MLP neural network can be an efficient choice for this purpose since the network deals with deviations and not the actual signals. For a typical MLP network with *m*, *n* and *r* neurons in the input, hidden and output layers respectively, and the hyperbolic tangent as the activation function, it can be shown that:

$$\hat{Y}(t) = V(t) \times d(t), \tag{9-9}$$

where $\hat{Y}(t) \in R^r$, $V(t) \in R^{r \times n}$ and $d(t) \in R^n$ are the output vector, output weight matrix and the decision vector respectively (see Appendix C). The k^{th} output of the neural network can therefore be written as:

$$\hat{y}_k(t) = V_k(t) \times d(t), \tag{9-10}$$

where $V_k(t)$ is the k^{th} row of the output weight matrix. This can be further expanded as:

$$\hat{y}_k(t) = V_k(t) \times \phi[W(t) \times X(t)], \tag{9-11}$$

where $W(t) \in R^{n \times m}$ and $X(t) \in R^m$ are the input weight matrix and the input vector of the neural network respectively. During backpropagation training, the entries of the input weight matrix are updated according to Equation (9-12):

$$\Delta w_{ji}(t) = \eta \times x_i(t) \times \frac{d\phi[a_j(t)]}{dt} \times \sum_{k=1}^{r} [v_{kj}(t).e_k(t)], \tag{9-12}$$

where $a_j(t)$ and $e_k(t)$ are the entries of the activation vector and the output error vector respectively. It can be seen in Equation (9-12) that every output error $e_k(t)$ has a direct impact on all the entries of the input weight matrix, and therefore, on all the entries of the output vector. This means that in an MLP neural network the outputs interact with one another and the error in each one affects the others.

A WAC Action network designed using MLP networks will therefore create unwanted interactions between the controllers, whereas the idea behind the WAC is to generate an auxiliary reference signals for each local controller based on the effect of only that controller on the cost-to-go function.

This issue can be solved by using a *Functional Link* (FNL) neural network. These neural network structures were originally proposed by Pao *et al.* (1992). Using

orthogonal mathematical functions Φ(.), a FNL expands the *m*-dimensional input space to the *M*-dimensional hyperspace, where $M \geq m$. These orthogonal functions, which can range from polynomials to trigonometric functions, form the bases for the FNL neural network. Figure 9.7 illustrates the schematic diagram of a FNL neural network, where *V* and λ(.) are the output weight matrix and the output activation function respectively. In this structure, the hidden layer is replaced by the Function Expansion block; therefore, the input weight links are removed as well. This prevents the output error of the k^{th} node to affect the signal generated by any of the other output layer neurons and therefore, there is no interaction between the different control signals. Instead, the WAC control signals try to independently control their corresponding local controller based on their individual effect on the overall cost-to-go function defined for the power system.

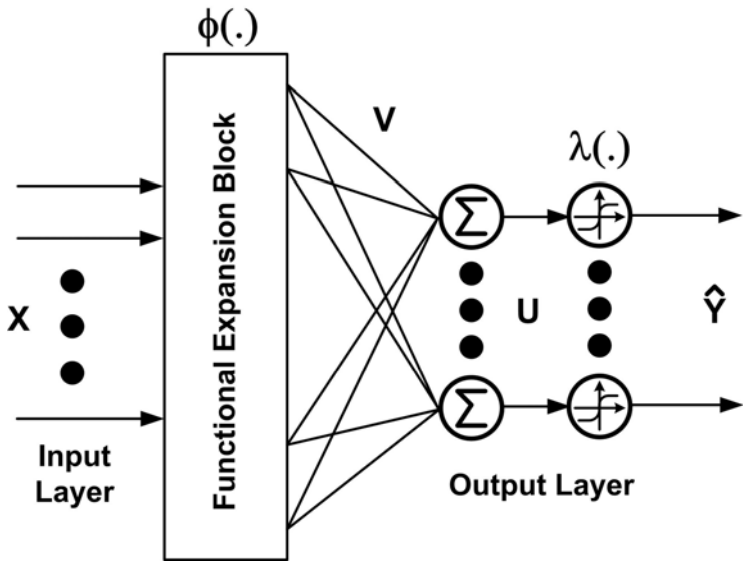

Figure 9.7. Schematic diagram of a FNL neural network.

The effect of the k^{th} output error $e_k(t)$ on the different entries of the neural network output vector can also be shown in mathematical terms. The k^{th} output node in a FNL neural network can be expressed as:

$$y_k(t) = \lambda[\sum_{j=1}^{M} v_{kj}(t) \times d_j(t)], \tag{9-13}$$

where $\lambda(.)$ is the output node activation function. Clearly, every link v_{kj} is only adjusted based on the k^{th} output node error signal, as shown in Equation (9-14):

$$\Delta v_{kj}(t) = \eta \times e_k(t) \times \lambda'[\sum_{j=1}^{M} v_{kj}(t) \times d_j(t)] \times d_j(t). \tag{9-14}$$

The details of a FNL neural network and its training algorithm are presented in Appendix C.

Figure 9.8 shows the schematic diagram of the Action network. A Chebyshev based FNL neural network is used where the first four Chebyshev functions are considered to be the activation (basis) functions[1]:

$$T_0(x) = 1.0,\ T_1(x) = x,\ T_2(x) = 2x^2 - 1,\ T_3(x) = 4x^3 - 3x. \tag{9-15}$$

The Chebyshev functions defined in Equation (9-14) are one dimensional; however, the Action network has three inputs, i.e., the speed deviations of the three generators. Therefore, the Chebyshev functions selected should be able to model the multi-dimensionality of the problem space. Various schemes are presented in the literature to address this issue. Some researchers have defined basis functions as *n* single dimensional functions for each input (Patra and Kot, 2002; Patra *et al.*, 1999), whereas other researchers have assumed that each basis function is constituted of the multiplication of

[1] The higher order Chebyshev polynomials can be generated by the recursive formula:
$T_{n+1}(x) = 2x.T_n(x) - T_{n-1}(x)$

the single dimensional basis functions (Yang and Tseng, 1996). A different approach is selected in this chapter that takes the first four Chebyshev functions as well as their combinations as the bases of the functional expansion module. In this way, the interdependence between the three inputs is taken into account as well.

Clearly, using more bases for the Function Expansion block can increase the accuracy of the Action network performance. However, this will in turn add to the computational intensity of the problem. Therefore, there should be a tradeoff between the performance of the FNL and the resultant computational complexity. To some extent, this problem is similar to determining the number of neurons in the hidden layer of a MLP or RBF neural network.

The bases for the FNL based Action neural network are listed in Equation (9-15), where T_i represents an ith order Chebyshev function. It has been observed in this study that increasing the bases in Equation (9-15) by considering more products of the single dimensional orthogonal basis functions does not drastically improve the performance of the WAC.

$$
\begin{aligned}
&1.0,\\
&T_1(\Delta\omega_1),\ T_2(\Delta\omega_1),\ T_3(\Delta\omega_1),\\
&T_1(\Delta\omega_2),\ T_2(\Delta\omega_2),\ T_3(\Delta\omega_2),\\
&T_1(\Delta\omega_3),\ T_2(\Delta\omega_3),\ T_3(\Delta\omega_3),\\
&T_1(\Delta\omega_1).T_1(\Delta\omega_2),\ T_1(\Delta\omega_1).T_1(\Delta\omega_3),\ T_1(\Delta\omega_2).T_1(\Delta\omega_3),\\
&T_1(\Delta\omega_1).T_1(\Delta\omega_2).T_1(\Delta\omega_3),\\
&T_1(\Delta\omega_1).T_2(\Delta\omega_2),\ T_1(\Delta\omega_1).T_2(\Delta\omega_3),\\
&T_1(\Delta\omega_2).T_2(\Delta\omega_1),\ T_1(\Delta\omega_2).T_2(\Delta\omega_3),\\
&T_1(\Delta\omega_3).T_2(\Delta\omega_1),\ T_1(\Delta\omega_3).T_2(\Delta\omega_2),\\
&T_2(\Delta\omega_1).T_2(\Delta\omega_2),\ T_2(\Delta\omega_1).T_2(\Delta\omega_3),\ T_2(\Delta\omega_2).T_2(\Delta\omega_3),\\
&T_2(\Delta\omega_1).T_2(\Delta\omega_2).T_2(\Delta\omega_3).
\end{aligned}
\tag{9-15}
$$

The schematic diagram of the Action network structure in Figure 9.8 shows a FNL neural network with Chebychev basis functions. It receives the values of the power system states at time steps (t), $(t-1)$ and $(t-2)$, and in turn generates the control signal at time (t). The output of the Action network will be added to the AVRs of the three generators and the STATCOM line voltage reference (Figure 9.5). The outputs of the Action network are clamped in a way that the terminal voltages of the generators or the voltage at bus 4 in Figure 9.3, where the STATCOM is connected to the power system, never go beyond the acceptable range of [0.95,1.05] p.u.

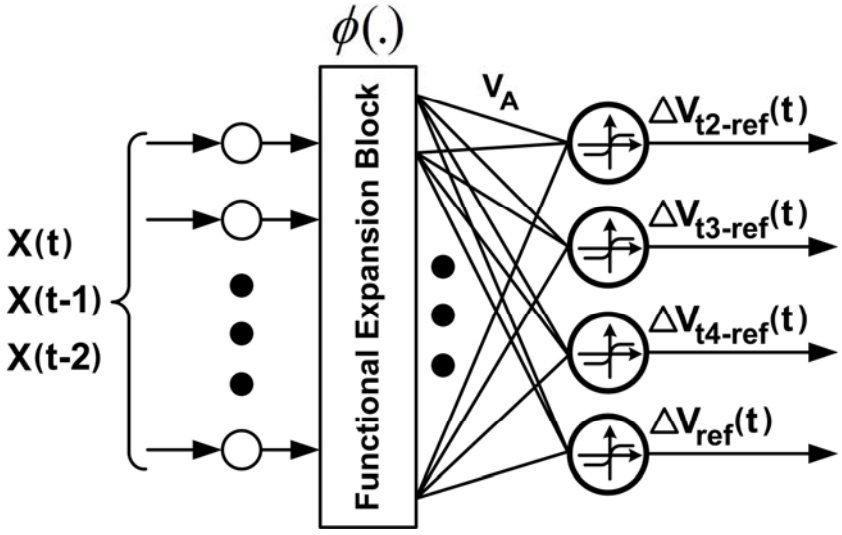

Figure 9.8. Schematic diagram of the WAC Action network.

It has been assumed in this study that the values of the system states at time (t) are available for the Critic and Action networks. However, in general this is not true due to the transport lags in sending/receiving data from across the network, and a Wide Area State Predictor such as the one introduced in Chapter 7 is necessary to predict the correct values of the plant states at time (t).

9.5. Neurocontroller Training

Figure 9.9 illustrates the schematic diagram of the WAC in the IEEE 12-bus power system.

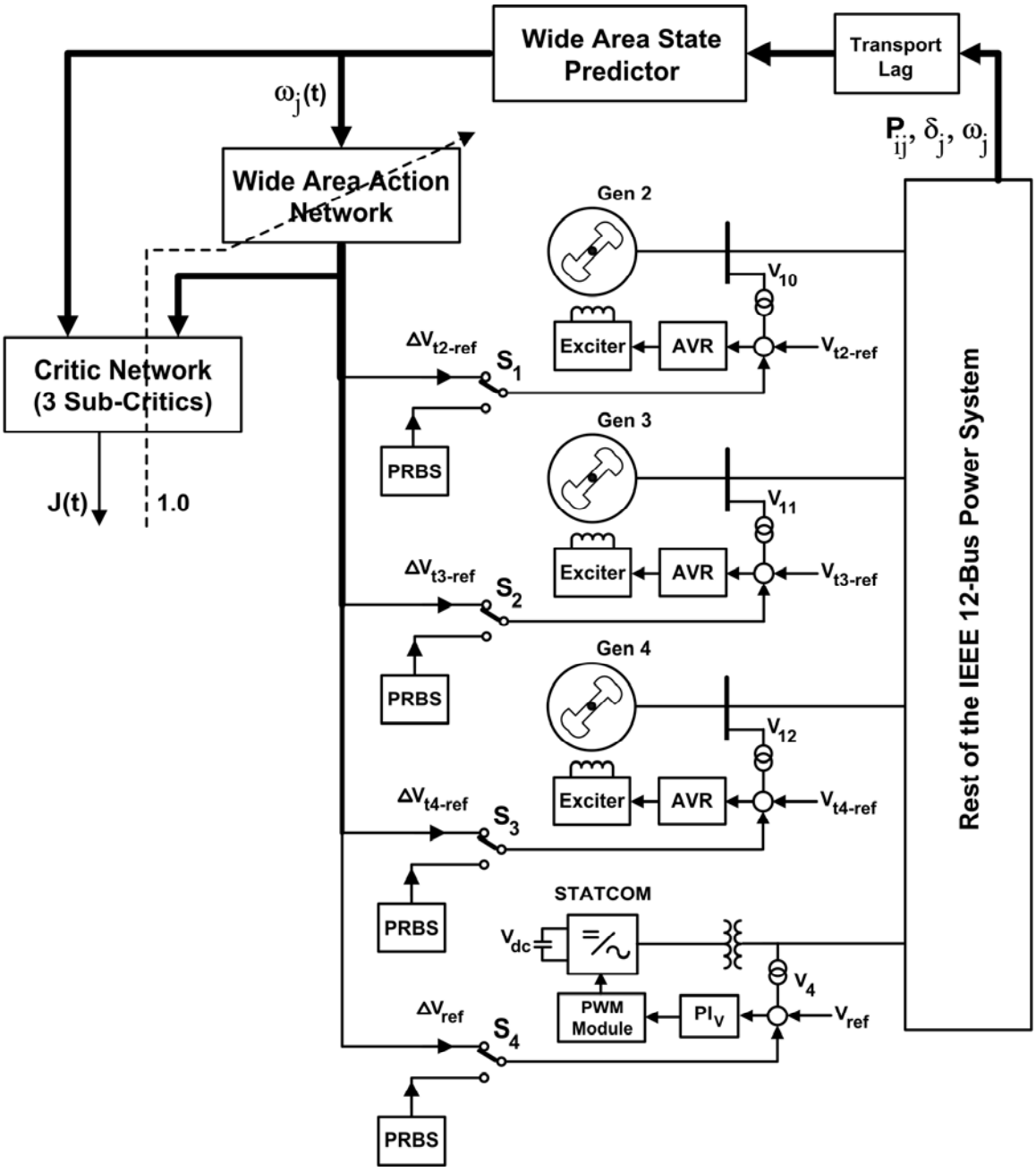

Figure 9.9. WAC in the IEEE 12-bus power system.

9.5.1. Critic Network Pre-Training

A period of forced training is first executed for training each of the three sub-Critic neural networks. During this stage, a PRBS disturbance is added to the reference signal of each one of the three AVR's as well as the PI_V line voltage reference as illustrated in Figure 9.9. Although these four PRBS disturbances have the same combination of frequencies (0.5, 1 and 2 Hz), they are not in phase with one another. The magnitudes of the PRBS disturbances are limited to $\pm 5\%$ of their corresponding steady state reference signals.

During the pre-training stage, the output of the Action network is not applied to the power system. Table 9.1 shows the procedure by which the sub-Critic networks are trained. The duration of each training cycle is 400 sec of simulation time. Only one operating point of the power system in Figure 9.3 is considered for the pre-training stage of the Critic network.

Table 9.1. Training cycles of the WAC Critic neural network.

Training Cycle #	Disturbance Applied to the Power System	Sub-Critics Undergoing Training
1	$\Delta V_{t2-PRBS}$	Networks # 1,3
2	$\Delta V_{t3-PRBS}$	Networks # 2,3
3	$\Delta V_{t4-PRBS}$	All Networks
4	$\Delta V_{ref-PRBS}$	All Networks
5	$\Delta V_{t2-PRBS}, \Delta V_{t4-PRBS}$	All Networks
6	$\Delta V_{t3-PRBS}, \Delta V_{t4-PRBS}$	All Networks
7	$\Delta V_{t2-PRBS}, \Delta V_{t3-PRBS}, \Delta V_{t4-PRBS}$	All Networks
8	$\Delta V_{t2-PRBS}, \Delta V_{t3-PRBS}, \Delta V_{t4-PRBS}, \Delta V_{ref-PRBS}$	All Networks

Figures 9.10 and 9.11 show some typical PRBS disturbances applied to the reference signals of one of the AVR's as well as the typical response of the a generator's rotor speed.

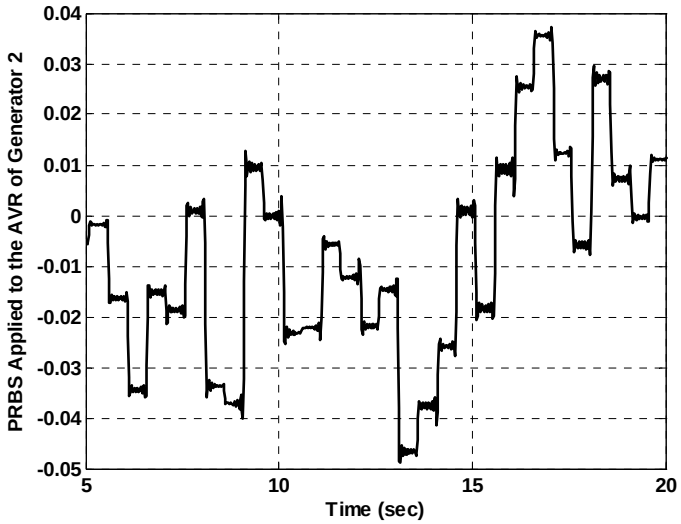

Figure 9.10. Typical PRBS disturbances applied to the AVR of generator 2 during the Critic network pre-training stage.

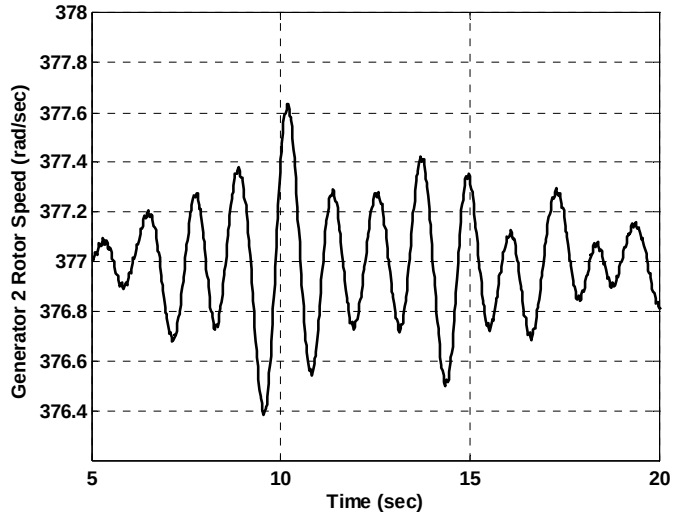

Figure 9.11. Rotor speed deviations of generator 2 during the Critic network pre-training stage.

The utility function and the output of the Critic network, along with the contribution of each sub-Critic network are depicted in Figures 9.12 and 9.13. The deviations in the generators' rotor speeds generate a positive utility function and therefore, a non-zero cost-to-go function. Each sub-Critic is now trained to estimate its corresponding cost-to-go function as illustrated in Figure 9.13.

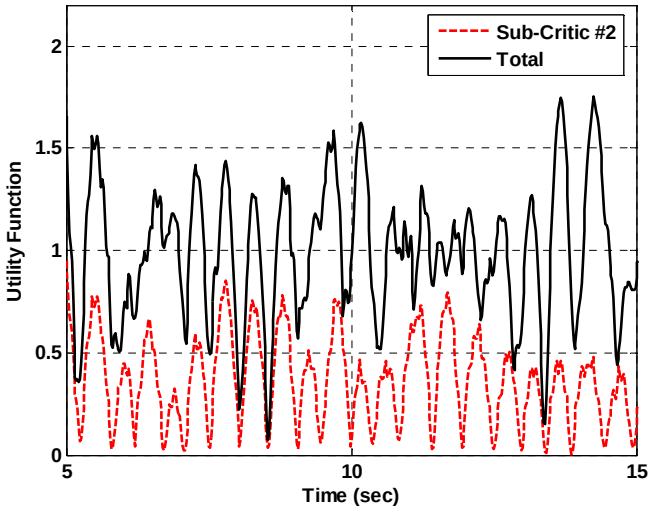

Figure 9.12. Utility function during the Critic network pre-training stage.

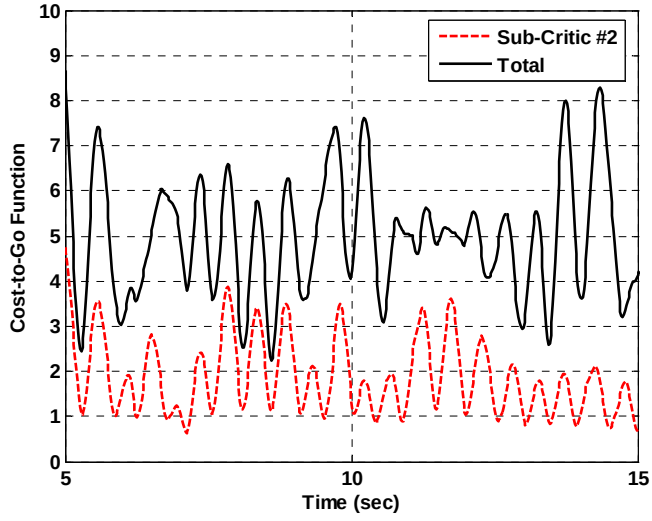

Figure 9.13. The overall cost-to-go function during the Critic network pre-training stage.

The Critic network pre-training starts with a low discount factor γ of 0.2 as in Equation (9-7), which is gradually increased to 0.8 as the training proceeds. This helps the weights of the sub-Critic networks to converge faster (see Chapter 6).

9.5.2. Action Network Pre-Training

Frequently it is necessary to pre-train the Action network, since the ACD controller is supposed to control the plant. In the case of supervisory level control, where the neural network is dealing with deviation signals, the Action network control loop can be closed and its output applied to the plant with a randomly initialized set of synaptic weights. The initial weights can be limited to a very small number so that they do not have much impact on the power system in the first stages of the training. As time goes by, the magnitudes of the Action network weights are increased and its outputs start impacting the power system.

However, it is possible to pre-train the Action network, using supervised learning, in order for the weights to converge to values that can help stabilize the power system right from the start. These values are the target signals that are defined by the user and are used in the supervised learning process. The weights after the pre-training stage are used as the starting point of the final training stage where the performance of the neurocontroller is improved towards an optimal nonlinear controller. If the target signals for the pre-training stage are chosen properly, the above procedure will expedite the overall learning process of the Action network.

A pre-training stage is executed for the WAC in this study, during which three power system stabilizers are installed on the three generators. The neurocontroller is trained to learn the dynamics of the three PSSs. The first three outputs of the WAC are related to

the three generators and the fourth output corresponds to the STATCOM line voltage controller. The latter is forced to follow the output provided by the linear external controller 3 in Chapter 8 (Figure 8.2). Therefore, the target signal for the Action network pre-training stage is defined as:

$$A^*(t) = [V_{PSS,2}(t), V_{PSS,3}(t), V_{PSS,4}(t), -k(\Delta\omega_3(t) + \Delta\omega_4(t))]^T . \quad (9\text{-}16)$$

Clearly, the above target vector is far from optimal; however, it provides a starting point solution for the Action network final training stage. Using a supervised learning scheme, the Action network weights are now trained for 500 seconds in order to follow the target vector defined in (9-16). The Action network is not controlling the plant at this stage; therefore, its outputs are not applied to the AVRs or the STATCOM.

9.5.3. Final Training Stage

With the Critic network and the Action network weights already converged during the pre-training stages, the ACD neurocontroller is now ready to control the power system. Therefore, the supervisory control loop can be closed by moving the switches S_1, S_2, S_3 and S_4 in Figure 9.9 from position 1 to position 2. At this stage, the Critic network is providing the appropriate training signal for the Action network.

The power system is now exposed to large scale faults and disturbances, such as three phase short circuits along different transmission lines and busses. The deviations of the generator rotor speeds create a non-zero utility function and therefore, a positive cost-to-go function $J(t)$ estimated by the Critic network. Both the Action and Critic networks undergo training in order to learn the dynamics of the power system during the large natural faults. An annealing learning rate scheme is used for both the Action and Critic. During this process, training starts with a learning rate of about 0.1 for the Critic and 0.01

for the Action, which gradually decreases to a value of 0.005 for the Critic and 0.001 for the Action network. This ensures that during the initial training stages the neural networks adapt themselves to the plant dynamics quickly, but as the learning process continues, they do not have a drastic reaction to any sudden changes in the plant dynamics. In this way the networks do not forget the previously learned information. During the simulations, the same discount factor and learning rate parameter are assumed for all the three sub-Critic networks. However, in general this is not necessary and the three neural networks can undergo training with different parameters. The training process continues at various operating conditions, with different faults and disturbances, until a good accuracy and acceptable performance is achieved by the ACD neurocontroller.

The overall process of training the ACD neurocontroller can be summarized below:

1. Training the three sub-Critic networks by applying PRBS disturbances, according to the procedure in Table 9.1,
2. Pre-Training the Action neural network, by a supervised learning scheme, based on the target signal stated in Equation (9-16),
3. Training the Action network by the training signals provided by the Critic network.

9.6. Simulation Results

The power system in Figure 9.3 is first simulated in the PSAT® toolbox in Matlab® software (Milano, 2006). The eigenvalue analysis results indicate that the system has three local oscillation modes corresponding to the three generators as shown in Table 9.2:

Table 9.2. Eigenvalue analysis of the IEEE 12-bus power system.

System Modes	Frequency	Participation Factors					
		δ_2	ω_2	δ_3	ω_3	δ_4	ω_4
-0.21±j.5.68	0.90 Hz	0.464	0.464	0.0002	0.0002	0.0215	0.0215
-0.32±j.7.47	1.18 Hz	0.0051	0.0051	0.412	0.412	0.0636	0.0636
-0.14±j.4.50	0.71 Hz	0.0169	0.0169	0.0692	0.0692	0.398	0.398

The frequencies in Table 9.2 show that each mode of oscillation in the power system in Figure 9.3 is related to one generator only; therefore, the power system does not have inter-area modes of oscillations between the synchronous generators. In order to damp out the local modes of oscillation corresponding to the generators, local stabilizers need to be designed that target these local modes.

The WAC is also trained according to the procedure stated in the previous section and its performance is compared with the uncompensated system and the system compensated using the local conventional PSS.

9.6.1. Case Study 1: A Three Phase Short Circuit at Bus 5

In the first test, a three phase short circuit occurs at bus 5. The fault is cleared after 100 *ms* and therefore, it does not permanently change the power system topology. Figures 9.14-9.15 illustrate the results. It can be seen that the WAC is only slightly more effective than the local PSS in damping out the speed oscillations.

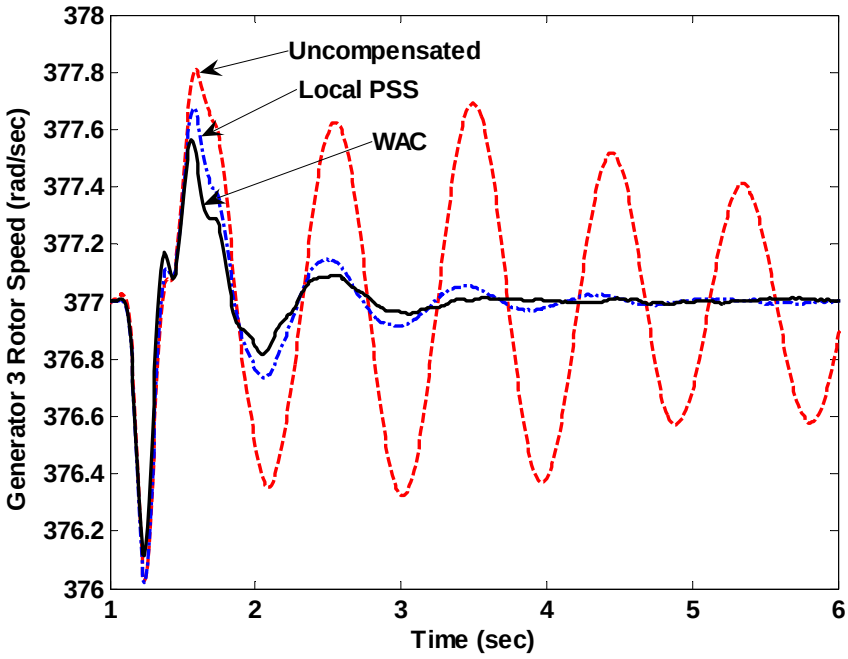

Figure 9.14. Rotor speed deviations of generator 3 during a 100 *ms* three phase short circuit at bus 5 (Figure 9.3).

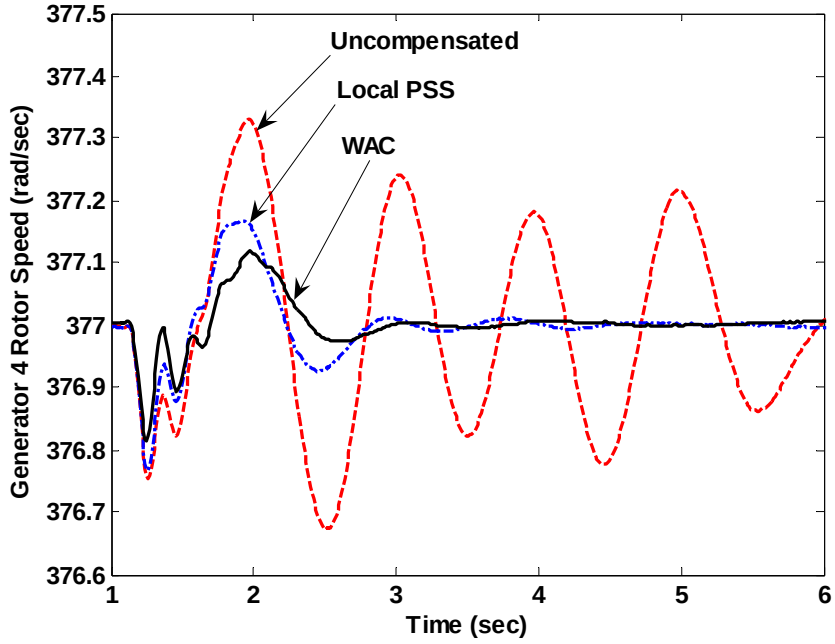

Figure 9.15. Rotor speed deviations of generator 4 during a 100 *ms* three phase short circuit at bus 5 (Figure 9.3).

9.6.2. Case Study 2: A Three Phase Short Circuit at the Middle of the Transmission Line 3-4

In the next test, a 100 *ms* three phase short circuit is applied at the middle of one of the parallel transmission lines connecting buses 3 and 4. The line is disconnected after the fault is cleared. Figures 9.16-9.18 compare the performances of the WAC and the local PSSs with an uncompensated system.

The figures show that the WAC is more effective than the case of the power system compensated with local PSSs.

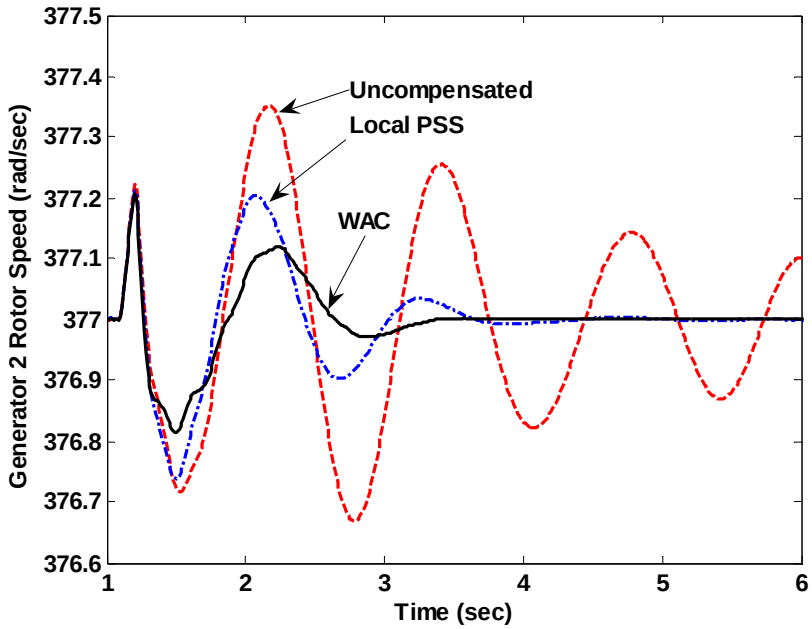

Figure 9.16. Rotor speed deviations of generator 2 during a 100 *ms* three phase short circuit at one of the transmission lines connecting buses 3 and 4 (Figure 9.3).

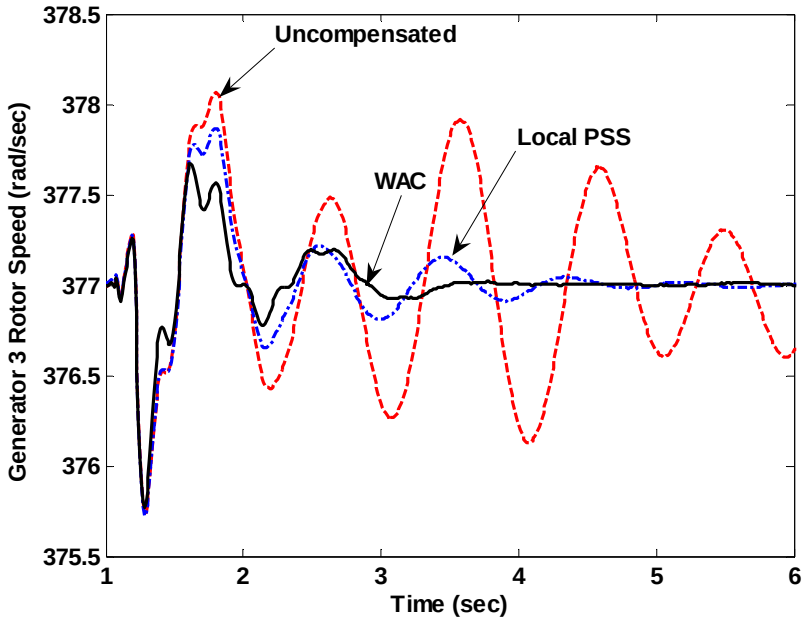

Figure 9.17. Rotor speed deviations of generator 3 during a 100 *ms* three phase short circuit at one of the transmission lines connecting buses 3 and 4 (Figure 9.3).

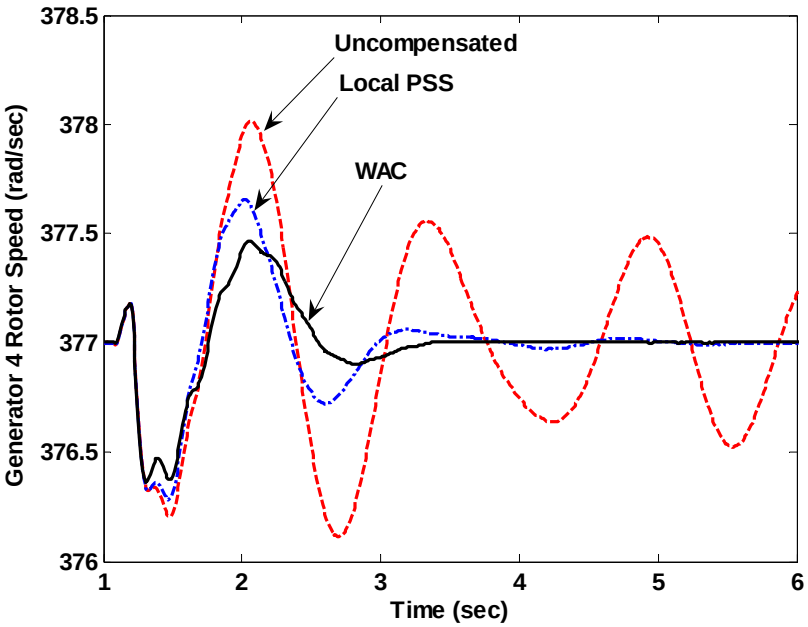

Figure 9.18. Rotor speed deviations of generator 4 during a 100 *ms* three phase short circuit at one of the transmission lines connecting buses 3 and 4 (Figure 9.3).

9.6.3. Case Study 3: Transmission Line 4-6 Disconnected

The next test investigates the effect of a major change to the topology of the power system by switching off a transmission line which connects buses 4 and 6. This changes the phase of the electric torque and therefore reduces the efficiency of the locally tuned stabilizers. Figures 9.19-9.22 contain some typical results. Figure 9.20 shows that for generator 4 the local PSS is still performing effectively; however, the WAC is considerably more effective for rotor speed deviations in generator 3 (Figure 9.19). This can be due to the fact that the dynamics of generator 3 are affected more by the topology change in the power system.

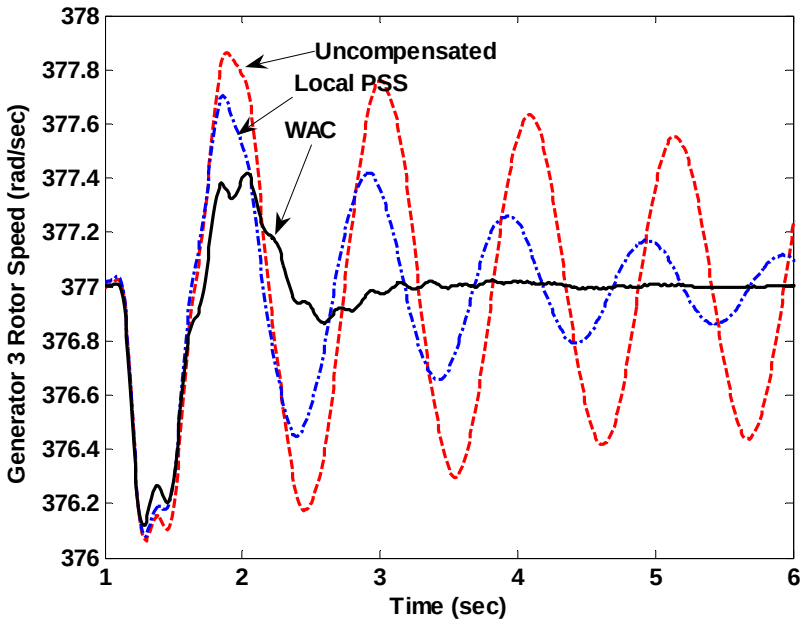

Figure 9.19. Rotor speed deviations of generator 3 when the transmission line connecting buses 4 and 6 (Figure 9.3) is disconnected.

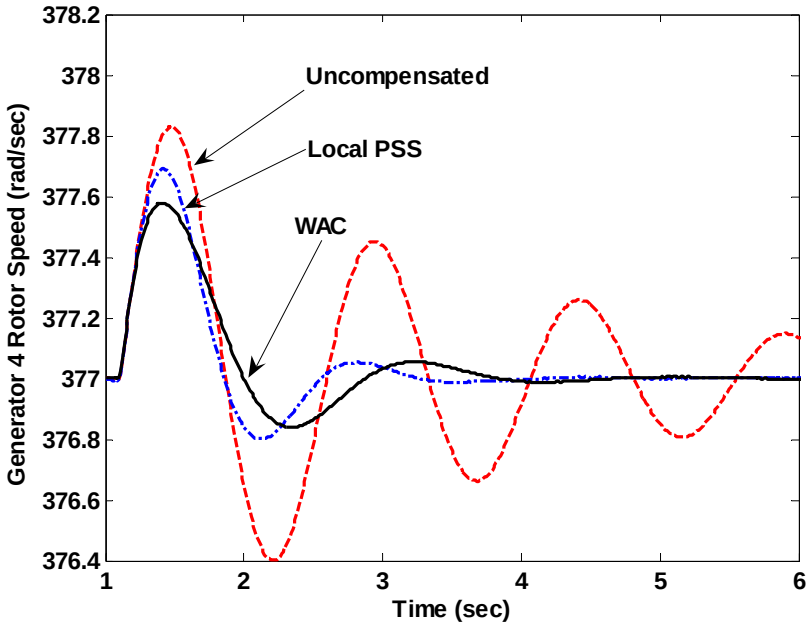

Figure 9.20. Rotor speed deviations of generator 4 when the transmission line connecting buses 4 and 6 (Figure 9.3) is disconnected.

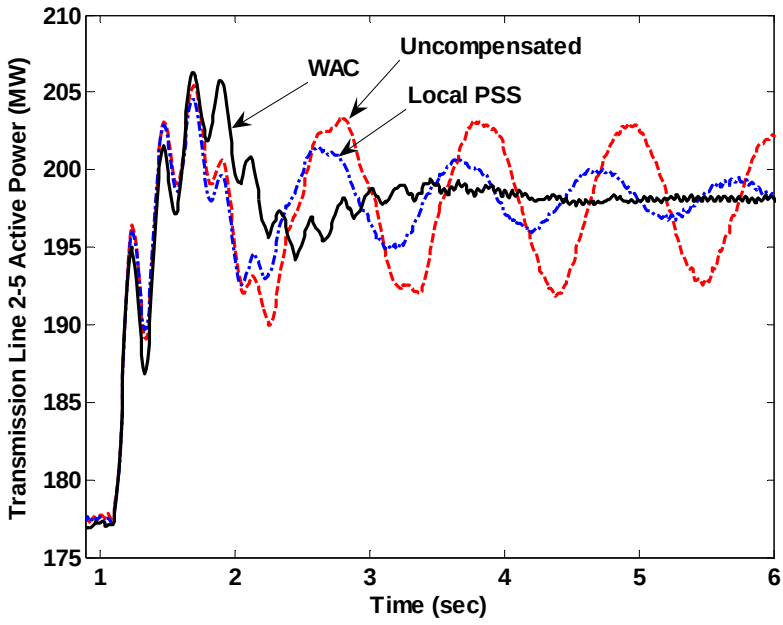

Figure 9.21. Active power flow through the transmission line 2-5 when the transmission line connecting buses 4 and 6 (Figure 9.3) is disconnected.

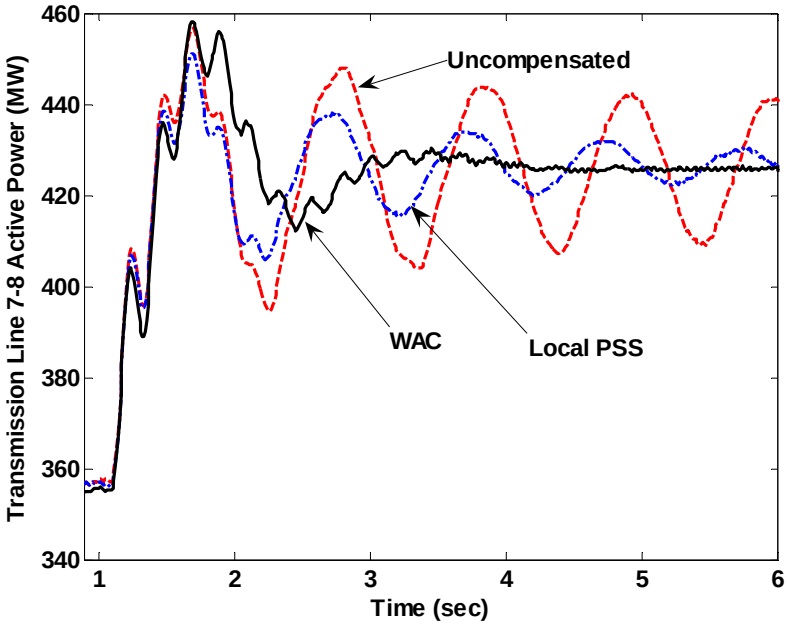

Figure 9.22. Active power flow through the transmission line 7-8 when the transmission line connecting buses 4 and 6 (Figure 9.3) is disconnected.

9.6.4. Case Study 4: Three Phase Short Circuit at the Middle of the Transmission Line 2-5

A three phase short circuit now occurs at the middle of the line connecting buses 2 and 5. The fault is cleared after 100 ms, but the line remains disconnected. Figures 9.23 and 9.24 show the rotor speed deviations of generators 3 and 4. Since the operating conditions and the topology of the power system change as a result of the fault, the performances of the local PSSs are slightly degraded. However, the WAC manages to restore the system to the steady state conditions in less time.

Simulation results also indicate that the WAC damps out the active power flow oscillations more effectively, with considerably less overshoot, than the local PSSs or the uncompensated system (Figures 9.25-9.27).

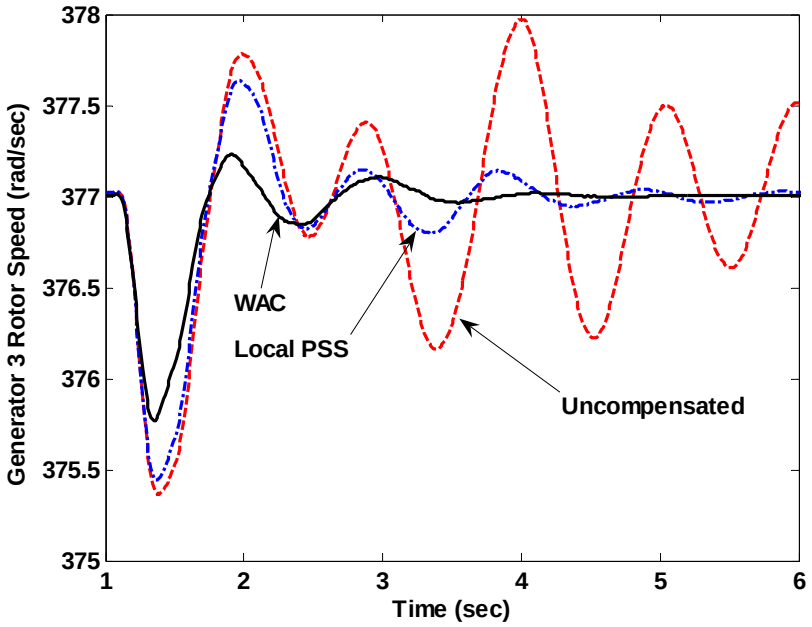

Figure 9.23. Rotor speed of generator 3 during a 100 *m*s three phase short circuit at the transmission line connecting buses 2 and 5 (Figure 9.3).

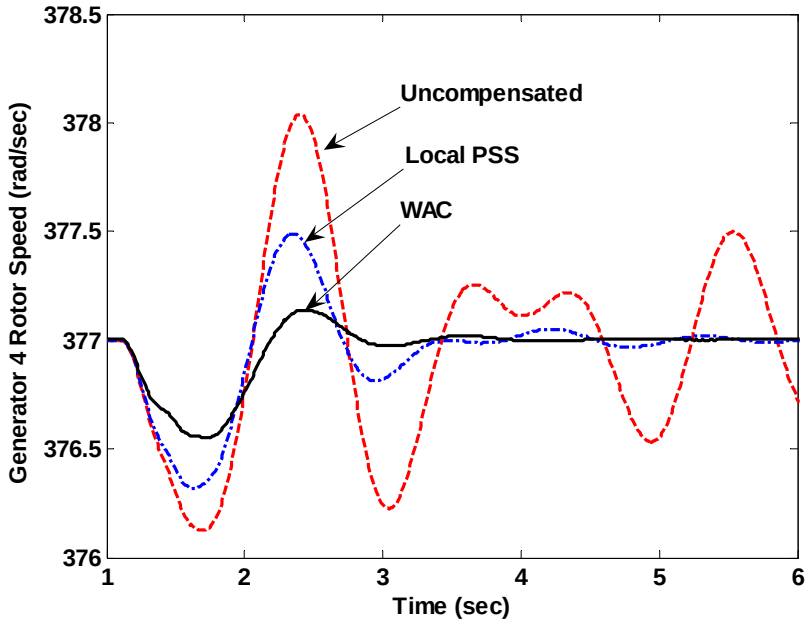

Figure 9.24. Rotor speed of generator 4 during a 100 *m*s three phase short circuit at the transmission line connecting buses 2 and 5 (Figure 9.3).

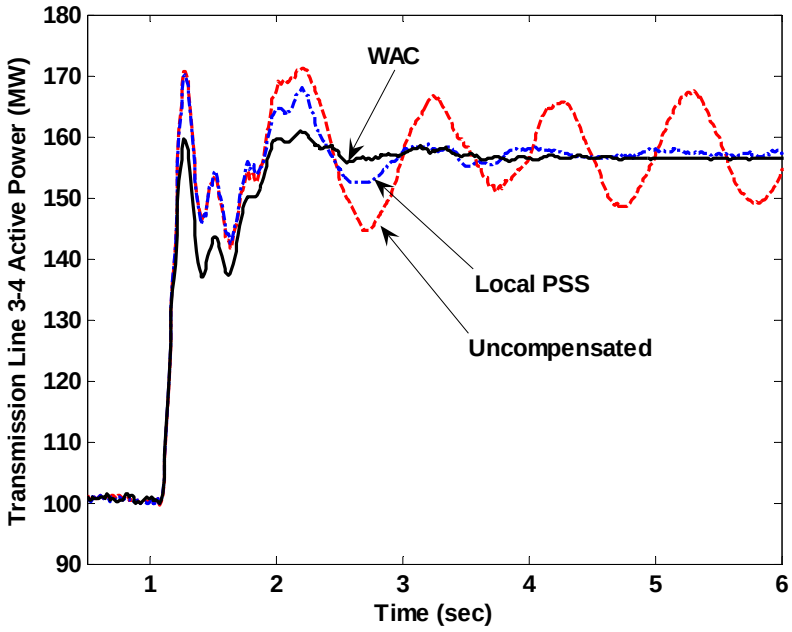

Figure 9.25. Active power flow through the transmission line 3-4 during a 100 *m*s three phase short circuit at the transmission line connecting buses 2 and 5 (Figure 9.3).

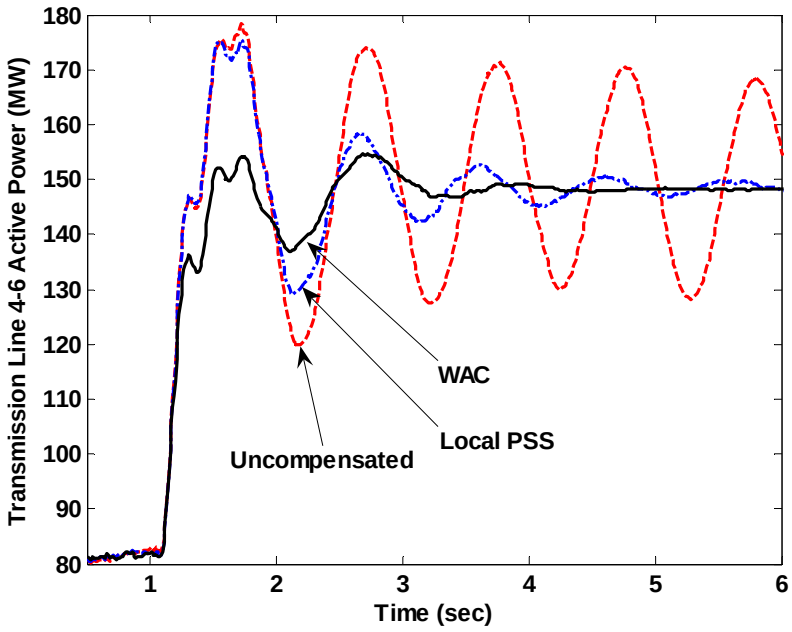

Figure 9.26. Active power flow through the transmission line 4-6 during a 100 *m*s three phase short circuit at the transmission line connecting buses 2 and 5 (Figure 9.3).

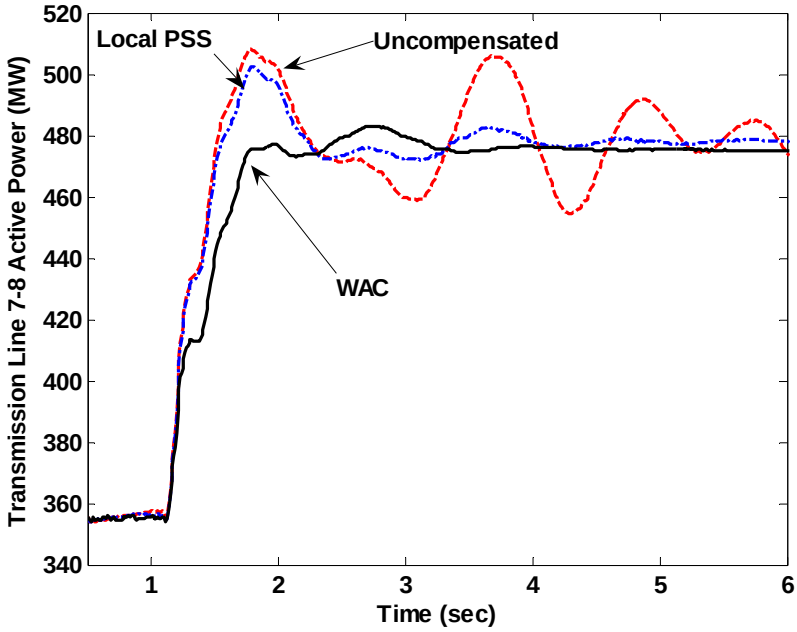

Figure 9.27. Active power flow through the transmission line 7-8 during a 100 *m*s three phase short circuit at the transmission line connecting buses 2 and 5 (Figure 9.3).

9.6.5. Case Study 5: Load Disconnected

In the next test, the operating condition of the power system is changed by changing the active power reference of generators 3 and 4. After the power system reaches the new steady state point, the shunt load in bus 4 is disconnected and the results are shown in Figures 9.28-9.32. The local PSS for generator 2 still manages to damp out the oscillations almost as fast as the WAC; however, the performances of the local stabilizers on generators 3 and 4 are noticeably degraded.

Figures 9.33-9.35 also illustrate the active power flow through some of the transmission lines in the system. Even though the operating conditions of the system are changed, the WAC still manages to damp out the oscillations effectively.

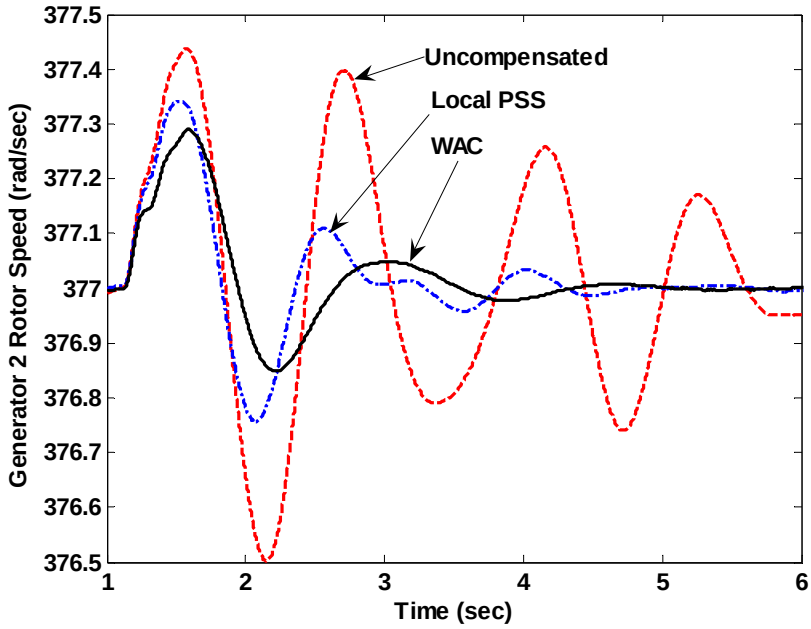

Figure 9.28. Rotor speed of generator 2 when the shunt load at bus 4 (Figure 9.3) is disconnected.

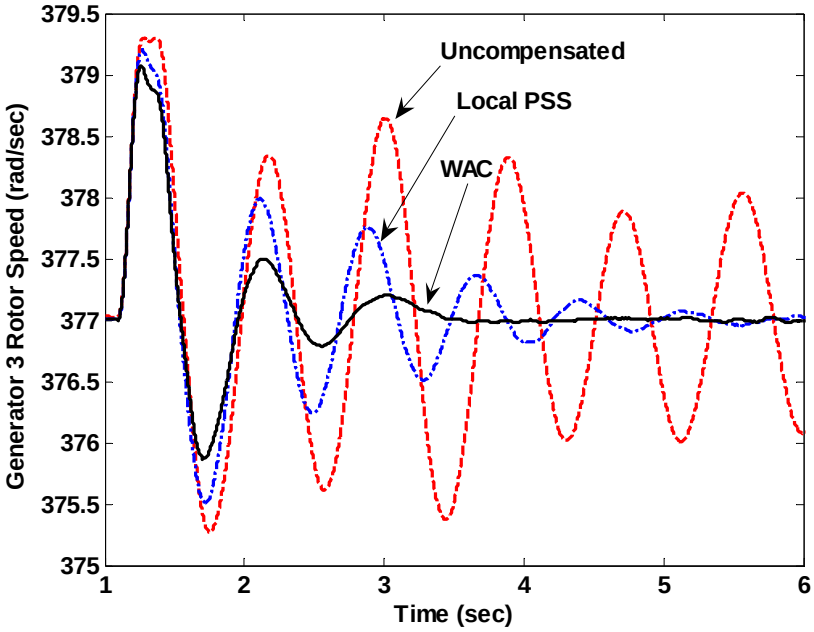

Figure 9.29. Rotor speed of generator 3 when the shunt load at bus 4 (Figure 9.3) is disconnected.

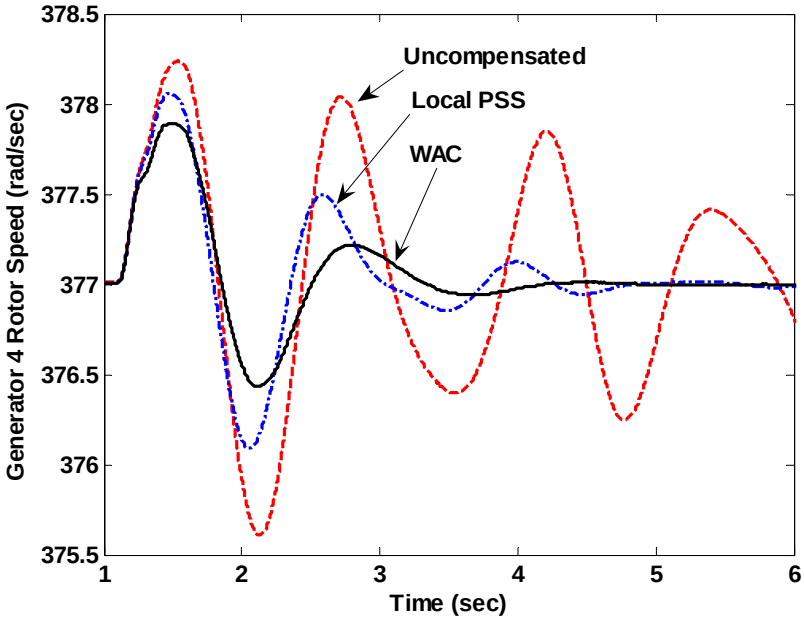

Figure 9.30. Rotor speed of generator 4 when the shunt load at bus 4 (Figure 9.3) is disconnected.

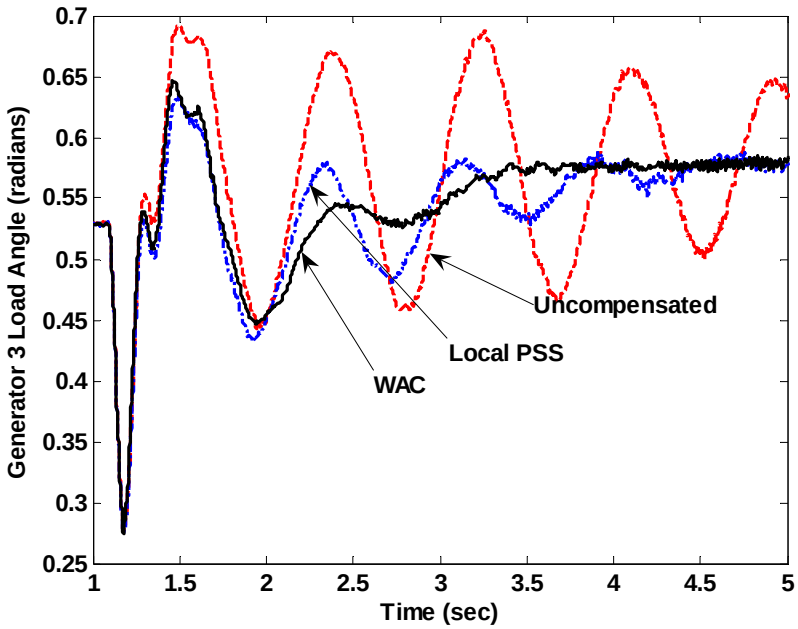

Figure 9.31. Generator 3 load angle when the shunt load at bus 4 (Figure 9.3) is disconnected.

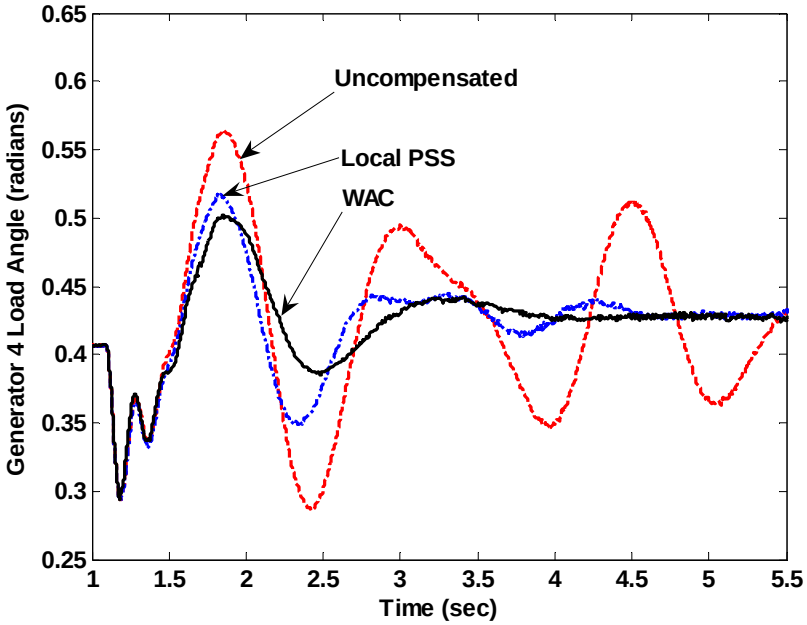

Figure 9.32. Generator 4 load angle when the shunt load at bus 4 (Figure 9.3) is disconnected.

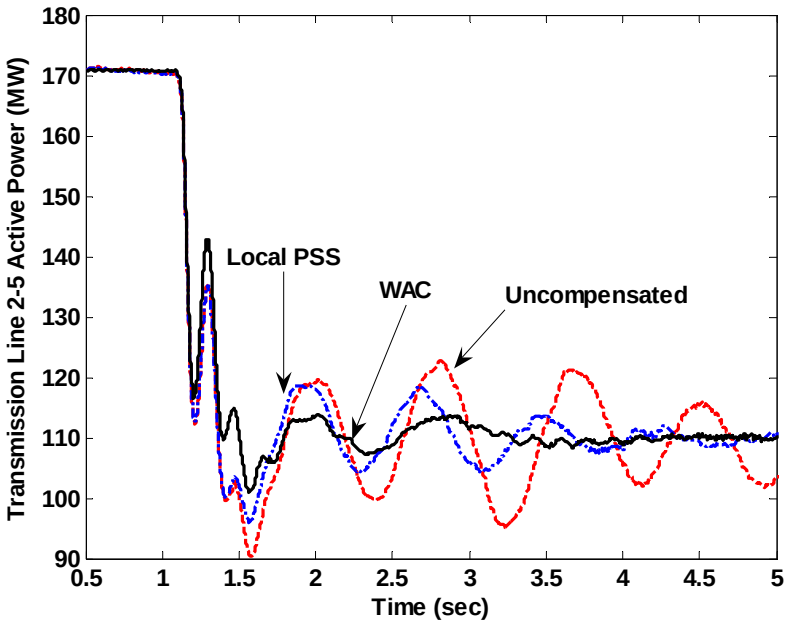

Figure 9.33. Active power flow through the transmission line 2-5 when the shunt load at bus 4 (Figure 9.3) is disconnected.

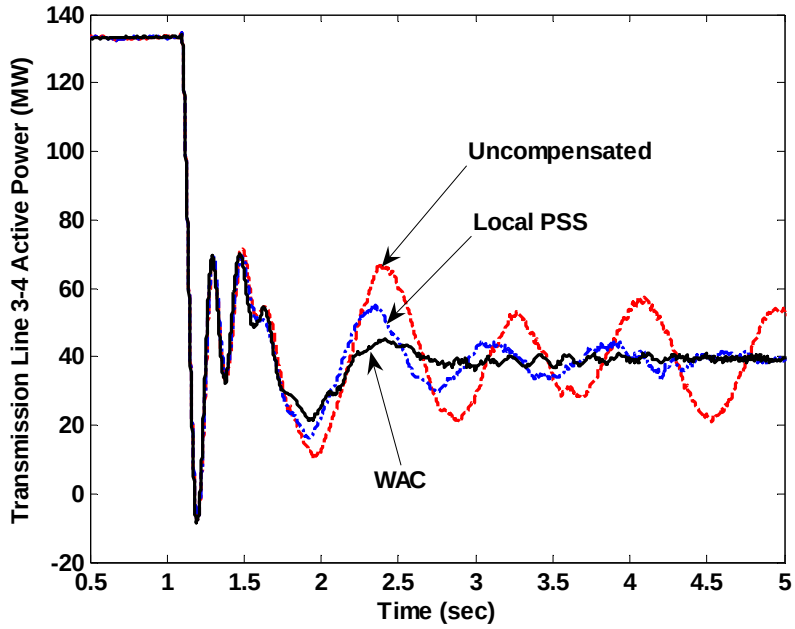

Figure 9.34. Active power flow through the transmission line 3-4 when the shunt load at bus 4 (Figure 9.3) is disconnected.

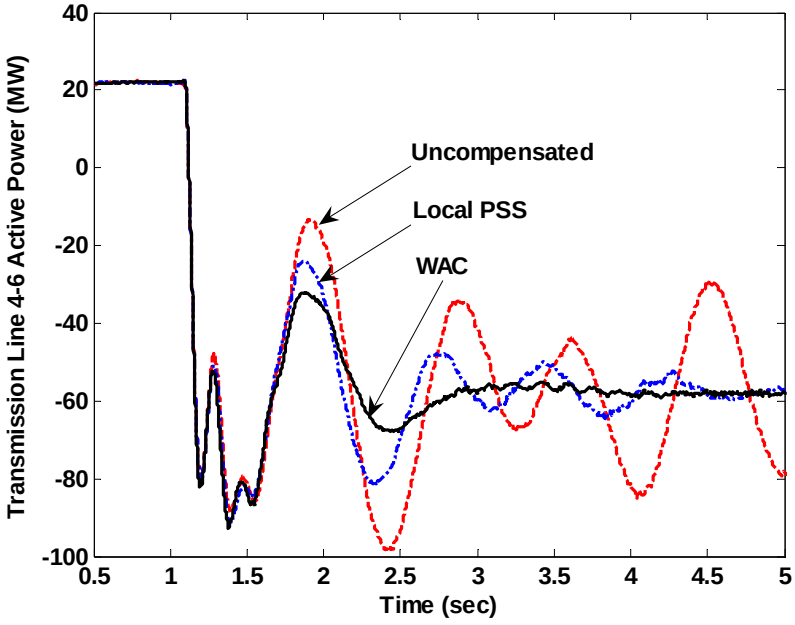

Figure 9.35. Active power flow through the transmission line 4-6 when the shunt load at bus 4 (Figure 9.3) is disconnected.

9.6.6. Performance Measurement

In this section, the performance of the IEEE 12-bus power system equipped with the ACD wide area controller is compared with the local PSSs as well as the uncompensated power system. A performance index is defined for each case study presented in sections 9.6.1 to 9.6.5 according to in Equation (8-1). Table 9.3 summarizes the results. The overall performance index of each control scheme in Table 9.3 is defined as in Equation (8-2). In the last row of the Table 9.3 the overall performance indices are normalized based on the overall performance index of the ACD wide area neurocontroller.

Table 9.3. Performance indices of the ACD wide area controller, the power system compensated with local power system stabilizers and the uncompensated power system.

Type of Test	Performance Index		
	Uncompensated System	Local PSS	WAC
Case Study 1	1.33	3.55	4.15
Case Study 2	1.17	2.28	2.94
Case Study 3	1.26	2.49	3.20
Case Study 4	0.93	1.78	2.41
Case Study 5	0.71	1.27	1.59
Overall Performance Index	0.204	0.404	0.516
Normalized Overall Performance Index	0.39	0.78	1

9.7. Conclusions

An adaptive critic designs based wide area neurocontroller has been designed in this chapter for hierarchical control of the IEEE 12-bus power system. It provides optimal auxiliary control signals for the automatic voltage regulators of the generators and the voltage control loop of the STATCOM. The wide area controller (WAC) proves to be more effective in providing system damping than the local PSSs designed for the respective generators. Several cases have been studied in order to investigate the efficiency of the proposed WAC at different operating conditions and system configurations.

CHAPTER 10

CONCLUSIONS, CONTRIBUTIONS AND RECOMMENDATIONS

10.1. Conclusions

A power system is a nonlinear non-stationary system that consists of various locally controlled components such as generators and FACTS devices. By regulating the amount of active and reactive power exchanged between these components and the network, they are able to improve the steady state voltage/power profile of the power system, as well as its dynamic and transient performance. While almost all current control techniques are sufficiently effective during steady state conditions, they are often far from optimal during large transient and small dynamic disturbances. That is when the controller needs to have a combination of speed and accuracy.

Linear PI controllers are traditionally used to locally control generators and FACTS devices. Nonlinear control techniques have been proposed in the literature in order to overcome the main disadvantage of the PI controllers, namely that their performance depends on the operating conditions and topologies of the power system. However, these nonlinear techniques in turn suffer from the disadvantage of requiring a mathematical model of the plant to be controlled.

This thesis investigated another alternative for solving the above issues; namely intelligent controllers. It focused on using a static compensator (STATCOM), which is a shunt connected FACTS device that can control the voltage at the point of connection to the power network. Adaptive and optimal controllers were designed using artificial neural networks and fuzzy logic based systems for local control of the STATCOM.

The design techniques employed in this thesis for implementing intelligent internal controllers for the STATCOM can be also applied to other types of shunt FACTS devices such as the SVC.

In addition to designing nonlinear internal controllers for the STATCOM, intelligent hierarchical controllers were implemented using adaptive critic designs theory in order to control a multimachine power system, including a STATCOM, from a supervisory level.

Figure 10.1 illustrates the outline of the thesis.

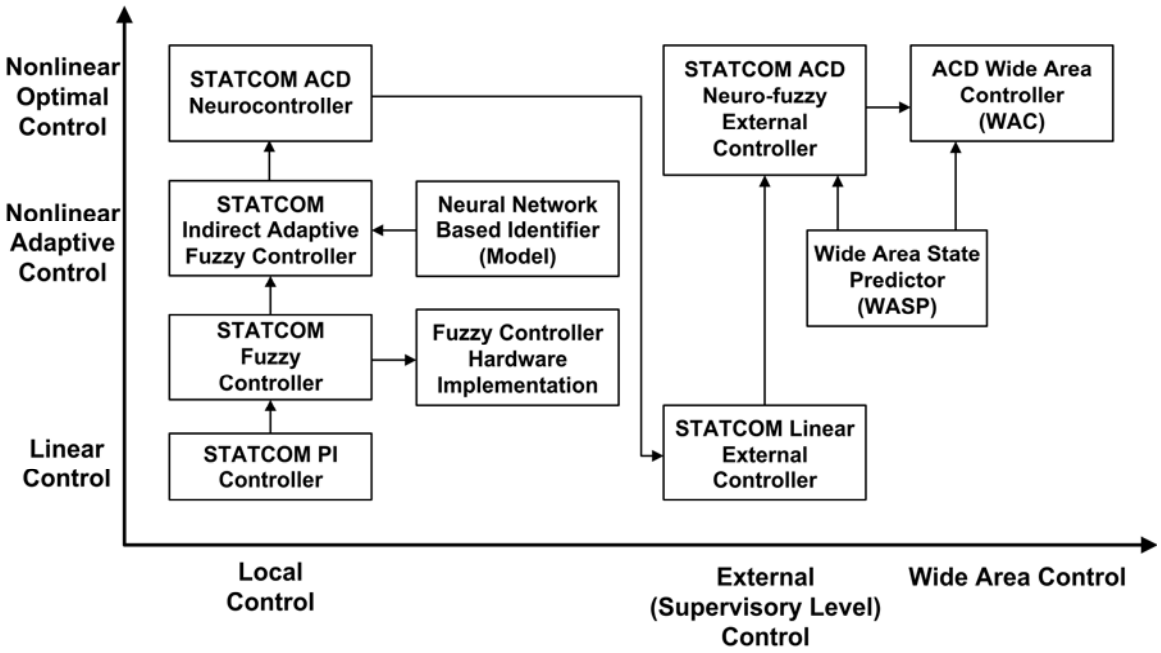

Figure 10.1. Outline of the thesis.

10.1.1. Identification of a Multimachine Power System

A radial basis function (RBF) neural network was designed in Chapter 4 to model a multimachine power system with a STATCOM. The centers of the RBF units were determined based on an offline clustering technique, after taking data samples at various operating conditions and during different tests. The output weights of the neural network

were updated online in order to give more flexibility to the neural network to deal more effectively with an operating condition that was far from its training points.

A step by step procedure provided in Chapter 4 can be applied for designing a neuroidentifier for any power system.

10.1.2. Indirect Adaptive Fuzzy Controller for a STATCOM

Chapter 5 proposed the first intelligent controller in this thesis for a STATCOM connected to a multimachine power system (Figure 4.1). The Takagi-Sugeno fuzzy inference mechanism was used for implementing the controller. The fuzzy controller was first implemented using shrinking span membership functions (SSMF) in order to give a smooth control trajectory to the controller. The performance of the fuzzy controller was then further improved by applying a model based adaptive scheme, in which the parameters of the controller were adaptively updated according to a pre-defined cost function in Equation (5-9), so that the controller minimizes this function at one step ahead. A neuroidentifier, as described in Chapter 4, was employed as the model of the power system, which predicted the power system outputs at one step ahead, thereby creating the appropriate cost function.

Simulation results were provided to show that the proposed indirect adaptive fuzzy controller was more effective than a conventional fixed-structure fuzzy controller during large scale as well as small scale disturbances. Both these techniques were proved to be more effective than a conventional PI controller, which is fine tuned at only one operating condition.

10.1.3. Hardware Implementation of a Fuzzy Controller for a STATCOM

Chapter 5 also presented the experimental results of a hardware implementation of a fuzzy controller for a STATCOM in the same power system as in Figure 4.1. The controller was designed and constructed on a DSP board, which interfaced to the multimachine power system implemented on a real-time digital simulator (RTDS®). The experimental results showed that the fuzzy controller, acting as a nonlinear gain scheduling controller, was more effective than a conventional PI controller during large scale disturbances. The fuzzy controller was also shown to restore the system into steady state conditions with considerably less control effort than the PI.

10.1.4. Adaptive Critic Designs Based Optimal Neurocontroller for STATCOM

A nonlinear optimal neurocontroller was proposed for the STATCOM in Chapter 6 that could further improve the system performance during faults and disturbances. Using adaptive critic designs (ACD) technique, the controller was able to overcome the "short-sightedness" of the adaptive controller proposed in Chapter 5. As a member of the approximate dynamic programming (ADP) family, the ACD technique can provide nonlinear optimal control over the infinite horizon of the problem in the presence of noise and uncertainties.

A Critic neural network was implemented and trained in order to estimate the cost-to-go function in the Bellman's equation. The trained Critic network was then used for training the Action neural network which was responsible for generating the optimal control signals for the STATCOM. The training of the two networks was repeated back and forth at several operating conditions. Once the training stage was finished and the weights of the two neural networks had converged, they were kept fixed and the

performance of the neurocontroller was tested during various tests applied to the power system.

Simulations were carried out on two multimachine power systems, namely the 9-bus system in Figure 4.1 and a 45-bus 10-generator system as a section of the Brazilian power network (Figure 6.1). The results indicated that the proposed nonlinear optimal neurocontroller outperforms both the adaptive fuzzy and the conventional PI controller in terms of the response speed/accuracy and the control effort provided by each one.

10.1.5. Wide Area State Predictor

The first challenge for any wide area control scheme is to have real-time access to the information from over its entire area in the presence of unavoidable transport lags. In Chapter 7, a neural network based structure was proposed for predicting the states of a multimachine power system in real-time.

Initially, a static transport lag of 100 *ms* was assumed for transmitting the information from the power system components to the wide area state predictor (WASP). Two identical RBF neural networks were used for performing a multiple step ahead prediction, in order to predict the power system states in real-time.

The performance of the WASP was further improved by introducing a third neural network: a multilayer perceptron (MLP) neural network to be trained as an autoencoder. Using a projection onto convex sets (POCS) algorithm, the autoencoder was trained to restore the missing sensor information using the healthy data in the event of a dynamic transport lag of more than 100 *ms* or single/multiple sensor failures.

Simulation results showed that using this technique, the WASP was able to predict the power system states in the presence of a static or a dynamic transport lag of up to 250 *ms*.

10.1.6. STATCOM Optimal Neuro-Fuzzy External Controller

Chapter 8 proposed an ACD based neuro-fuzzy external controller for a STATCOM in the IEEE 12-bus benchmark power system in Figure 8.1. The external controller received the rotor speed deviations from two of the generators neighboring the STATCOM and, based on the fuzzy rule-base, generated the required control signal in the form of an auxiliary reference signal for the line voltage control loop of the STATCOM. The performance of the neuro-fuzzy controller was optimized by introducing an action dependent heuristic dynamic programming (ADHDP) Critic neural network that estimated the cost-to-go function associated to the pre-defined utility function in Equation (8-7).

After training the Critic network and the fuzzy controller, the weights of the neuro-fuzzy controller were fixed and its performance was evaluated during several tests applied to the power system. The simulation results indicated that the proposed optimal neuro-fuzzy external controller was more effective in improving the dynamic stability of the power system compared to a conventional linear scheme as well as an uncompensated system.

10.1.7. Optimal Wide Area Controller

Chapter 9 extended the work in Chapter 8 by introducing a hierarchical controller for controlling the three generators and the STATCOM in the IEEE 12-bus benchmark power system. The wide area controller (WAC) was implemented based on the ACD

technique. A Critic neural network was designed using the ADHDP scheme, in order to estimate the cost-to-go function corresponding to the overall performance of the power system. The training signals provided by the Critic network were used to update the weights of the Action network, which in turn generated the optimal control signals for the generators and the STATCOM. The Action network was implemented using a functional link (FNL) neural network in order to avoid the unwanted interactions between the different control signals generated by the WAC.

The WAC provided auxiliary control signals for the generators' AVRs and the STATCOM voltage control loop. Simulation results were presented to show that the WAC was more effective than the local power system stabilizers (PSS) in improving the power system stability during small scale and large scale disturbances.

10.2. Contributions

The contributions made by this study are listed as follows:

- An elaborate step-by-step procedure was presented (Chapter 4) for implementing a neural network based identifier for any typical power system using RBF neural networks. This algorithm included the details for the selection of the neural network size, its complexity analysis, the size of the training data, training procedure and cross-validation testing.
- A detailed explanation was provided (Chapter 5) on implementing a fuzzy controller with fixed structure or with an adaptive structure using the indirect adaptive control scheme.

- The performance of an adaptive Takagi-Sugeno fuzzy controller was compared (Chapter 5) with those of a fixed structure fuzzy controller and a linear PI controller for a STATCOM in a 9-bus 2-generator multimachine power system.
- The fixed structure fuzzy controller was also built on a DSP board and its performance was validated (Chapter 5) by connecting it to a multimachine power system implemented in real-time on a RTDS®.
- An optimal ACD based neurocontroller was designed (Chapter 6) for a STATCOM and its performance was compared with a conventional PI controller on two multimachine power systems: the 9-bus 2-generator system and the 45-bus 10-generator section of the Brazilian power network.
- A neural network based Wide Area State Predictor was designed (Chapter 7) for predicting the power system states in real-time given static or dynamic transport lags of up to 250 *ms*.
- A detailed explanation was provided (Chapter 8) for designing an optimal neuro-fuzzy controller using ACD technique.
- An ACD based external neuro-fuzzy controller was implemented (Chapter 8) to control the STATCOM from a supervisory level. By generating the appropriate control signal in the form of the auxiliary reference signal, the external controller enabled the STATCOM to improve the dynamic stability of a multimachine power system.
- An ACD based optimal neurocontroller was designed (Chapter 9) for wide area control of the IEEE 12-bus benchmark power system. The controller provided

hierarchical control by sending auxiliary reference signals to the generators' AVRs and the STATCOM voltage references.

- The effectiveness of the FNL neural networks was analyzed (Chapter 9) for preventing the unwanted interactions between the control signals in a wide area control structure.

Most of the work in this thesis has already been published or accepted for publication as two journal papers (Mohagheghi *et al.*, 2006a, 2006f) and sixteen conference papers (Mohagheghi *et al.*, 2006b, 2006c, 2006d, 2006e, 2006g, 2005a, 2005b, 2005c, 2005d, 2005e, 2004a, 2004b, 2004c, 2003a, 2003b, 2003c).

10.3. Recommendations

Some recommendations can be made for continuing studies in this research field. These can be classified into two groups, namely theoretical/simulation and experimental recommendations.

10.3.1. Recommendations for Theoretical/Simulation Work

The theoretical grounds for designing optimal local controllers for power system components have matured throughout this study. The approach discussed in Chapters 6, 8 and 9 can be applied to any FACTS device or synchronous generator for designing a nonlinear controller capable of optimal control in the infinite horizon of the problem. However, designing a wide area controller for a larger power system still needs further investigation. Issues such as efficient training techniques for the Critic and Action networks, determining the appropriate structure (single Critic/controller or multiple Critics/controllers) and defining the most efficient utility functions are among the points that need further research.

In addition, this thesis has only focused on designing intelligent controllers (local or wide area) for the synchronous generators and the STATCOM. Investigating the possibility of extending the wide area control to other components in the power system, such as relays, passive compensators, transformer taps and suchlike might lead to interesting results.

10.3.2. Recommendations for Experimental Research

The implementation of the fuzzy controller in hardware and evaluating its performance on a real-time basis was the first step towards showing the practicality of the proposed intelligent controllers in power systems. This can be further extended by implementing the optimal neurocontroller for local control or wide area control in a multimachine power system.

APPENDIX A

MODELING A STATIC COMPENSATOR IN PSCAD/EMTDC®

A.1. Introduction

The static compensator (STATCOM) is modeled in detail in the PSCAD/EMTDC® environment. In this Appendix the characteristics of the components used for modeling the STATCOM, its switches, the RC snubber circuit and the step-up transformer are presented.

A.2. STATCOM Model

Figure A.1 shows the PSCAD model for a STATCOM. The device has a dc capacitor, six switches along with their corresponding RC snubber circuits and the reverse parallel diodes, the step-up transformer and the output filter. The dc link capacitor and the output filter capacitor are considered to be 300 μF and 2 μF respectively.

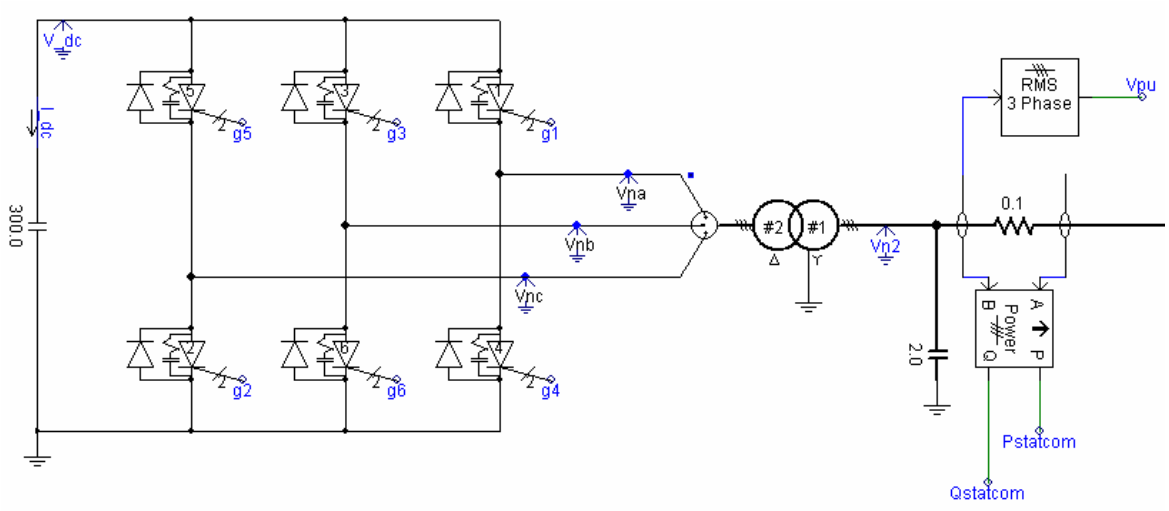

Figure A.1. PSCAD/EMTDC® model for a STATCOM.

A.2.1. Switch Model

Gate turn-off thyristors (GTO) are used for the STATCOM switches and their parameters appear in Table A.1. Also, Table A.2 presents the parameters of the reverse parallel diodes.

Table A.1. GTO parameters

Switch Parameter	Actual Value
Thyristor on resistance	0.005 Ω
Thyristor off resistance	1.0e8 Ω
Forward voltage drop	0 V
Forward breakover voltage	1.0e5 kV
Reverse withstand voltage	1.0e5 kV
Minimum extinction time	0 s
Snubber resistance	5000 Ω
Snubber capacitance	0.05 μF

Table A.2. Reverse diode parameters

Diode Parameter	Actual Value
Thyristor on resistance	0.005 Ω
Thyristor off resistance	1.0e8 Ω
Forward voltage drop	0 V
Forward breakover voltage	1.0e5 kV
Reverse withstand voltage	1.0e5 kV
Minimum extinction time	0 s

A.2.2. Step-Up Transformer

The output of the inverter is stepped up to the power system voltage level by using a step-up Δ-Y three phase transformer. The parameters of the transformer are given in Table A.3. Figure A.2 illustrates the saturation curve of the transformer.

Table A.3. Transformer parameters

Transformer Parameter	Actual Value
Three phase power	200 MVA
Rated voltage	50:230 kV
Base operation frequency	60 Hz
Positive sequence leakage reactance (L_1)	0.1 p.u
Air core reactance (L_{ac})	0.2 p.u
In-rush decay time constant	1.0 s
Knee voltage	1.25 p.u
Magnetizing current	1.0 %

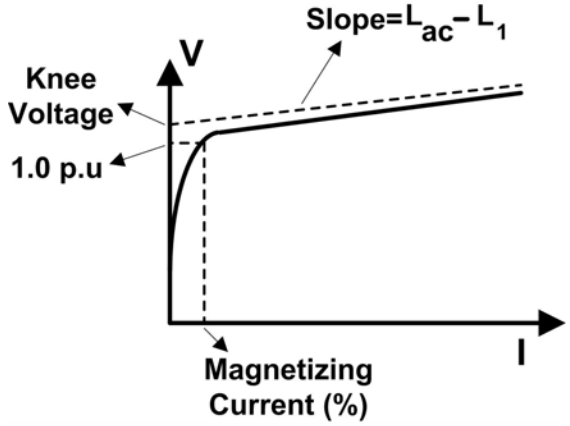

Figure A.2. Saturation curve of the transformer.

A.3. PWM Module

Once the appropriate modulation index and the phase shift signals are generated by the STATCOM controller, they are fed into the STATCOM PWM module. This block in turn generates the gate signals for the inverter switches. A sine-triangle PWM method is employed in this study. The first component of the PWM module is the triangular function generator synchronized with the ac system (Figure A.3). It consists of a phase locked loop (PLL), which is 3-phase and PI controlled, and generates a ramp signal "theta" that varies between 0 and 360° and is synchronized or locked in phase to the input voltage V_{na}. A frequency modulation ratio m_f of 15 is used for the inverter, therefore creating a triangular waveform with the frequency of 900 Hz.

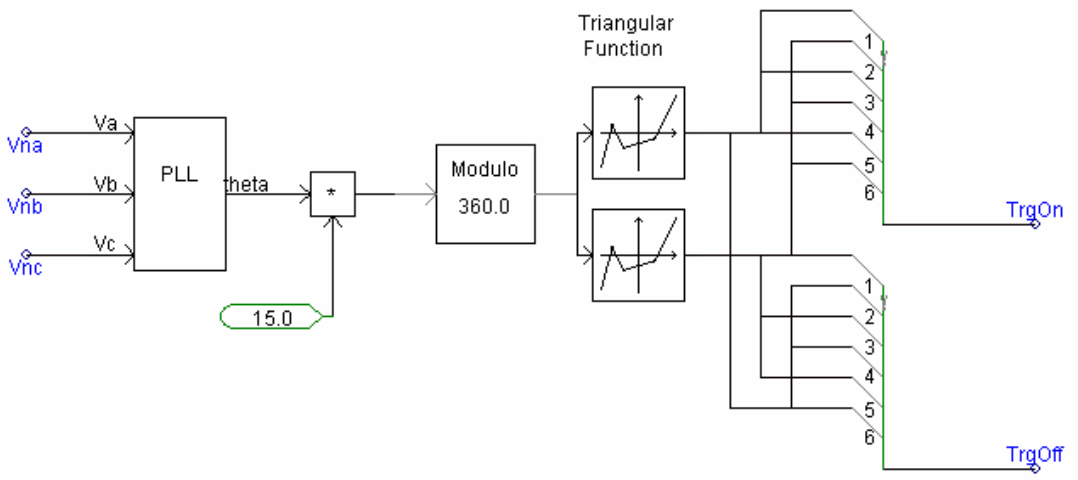

Figure A.3. PSCAD/EMTDC® model for the PWM module triangular function generator.

The triangular function block generates the triangular function with the desired frequency. The signals TrgOn and TrgOff in Figure A.3 are sent to the PWM pulse

generator in PSCAD in order to generate the turn-on and turn-off gate signals for the inverter switches.

The outputs of the blocks denoted "Triangular Function" in Figure A.3 are two triangular functions that are 180° out of phase. Clearly, the triangular function used for triggering on switches 1, 2 and 4 (Figure A.1) is the same as the function used for triggering off switches 1, 3 and 5, and vice versa. This is illustrated in Figure A.3.

The triangular functions are compared with the appropriate sinusoidal waveforms. Figure A.4 illustrates the block diagram for generating the sine waveforms. It consists of a second PLL for synchronizing the waveforms with the ac system. The PLL generates six output sinusoidal functions at its output, each one with a 60^o phase lag with respect to the previous one. Each of these output signals is shifted by $(\alpha - 30^o)$ degrees, where α is the inverter output phase shift signal and the 30° phase lag is due to the transformer Δ-Y configuration. The magnitude of the final sine wave is multiplied by the modulation index m_a.

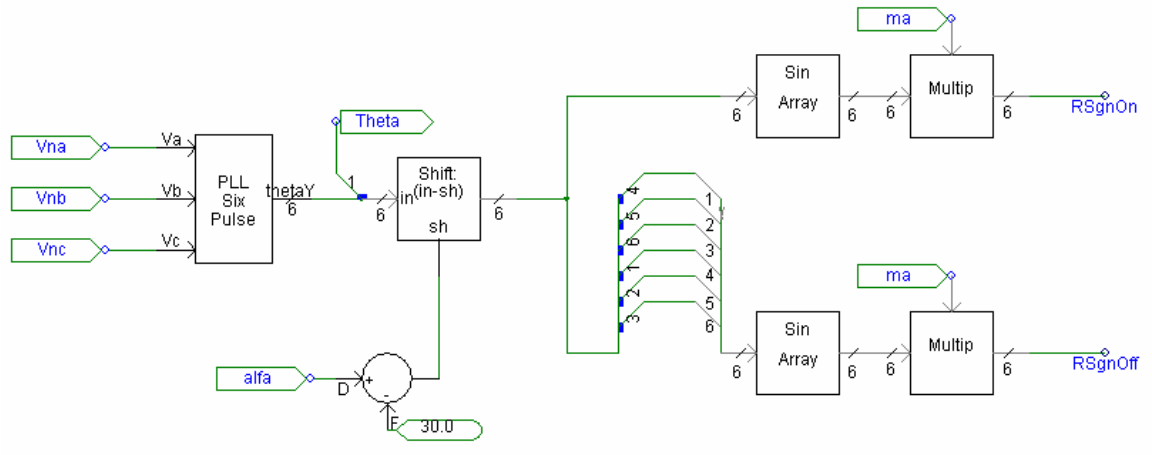

Figure A.4. PSCAD/EMTDC® model for the PWM module sine wave generator.

Clearly, the switches on the same inverter leg should not be conducting at the same time. Therefore, the sinusoidal functions for turning off switches 1, 2, 3, 4, 5 and 6 need to be the same as the sinusoidal functions for turning on switches 4, 5, 6, 1, 2 and 3 respectively. This logic is taken into account in Figure A.4.

The signals TrgOn and TrgOff in Figure A.3 are compared with the two signals RSgnOn and RSgnOff in Figure A.4 in order to generate the firing pulses for the inverter switches. This takes place in the last part of the PWM module, which is the "fp-int" block in PSCAD. This component returns the firing pulse and the interpolation time required for an interpolated turn-on (or turn-off) of the GTOs. The outputs of this component are in the form of a two-element array. The first output element is a 0 or 1 that represents the actual gate control pulse. The second is the information regarding the interpolated switching time.

EMTDC is a fixed time step transient simulation program and therefore, the time step is chosen at the beginning of the simulation, and remains constant thereafter. Due to the fixed nature of the time step, network events such as a fault or thyristor switching, can occur only at these discrete instants of time (if not corrected). This means that if a switching event occurs directly after a time step interval, then the actual event will not be represented until the following time step. This phenomenon can introduce inaccuracies and undesired switching delays. EMTDC uses an interpolation algorithm to find the exact instant of the event if it occurs between time steps. This is much faster and more accurate than reducing the time step and it allows EMTDC to accurately simulate any switching event, while still allowing the use of a larger time step.

The schematic diagram of the “fp-int” block is shown in Figure A.5. The signal “block” in Figure A.5 allows the user to enable the STATCOM at the desired point in time.

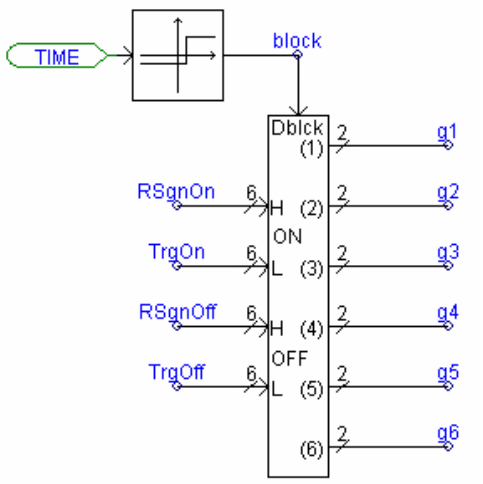

Figure A.5. PSCAD/EMTDC® model for the PWM module pulse generator.

APPENDIX B

MULTIMACHINE POWER SYSTEM PARAMETERS

B.1. Introduction

Three different multimachine power systems are simulated in the different chapters of this study: a 9-bus multimachine power system (Chapters 4-6), the IEEE 12-bus benchmark power system (Chapters 7-9) and a 45-bus 10-generator power system as a section of the Brazilian power grid (Chapter 6). Unless mentioned otherwise, all the simulations are carried out in the PSCAD/EMTDC® environment, which is a general-purpose time domain simulation tool for studying transient behavior of electrical networks. In all the above three multimachine power systems, all the generators are modeled in detail, with the dynamics of the governor, automatic voltage regulator (AVR) and exciter taken into account. The transmission lines are modeled using the π-model, without omitting the shunt admittance.

This Appendix presents the technical details and the parameters of the three multimachine power systems. In all the following sections, the multiple rotating masses of the generator and turbines are treated as one single rotating mass for each machine. Also, all the synchronous generators are modeled as salient pole machines; therefore, the q-axis air gap is large enough to avoid saturation in the q-axis direction. Moreover, the saturation effect on the d-axis is ignored for all the generators.

B.2. 9-Bus Multimachine Power System

Figure B.1 illustrates the 9-bus multimachine power system (Anderson and Fouad, 1994), which consists of nine buses with two generators and an infinite bus.

B.2.1. Generator Data

The parameters of the synchronous generators are presented in Table B.1. Both machines are considered to have one damper winding in the q-axis.

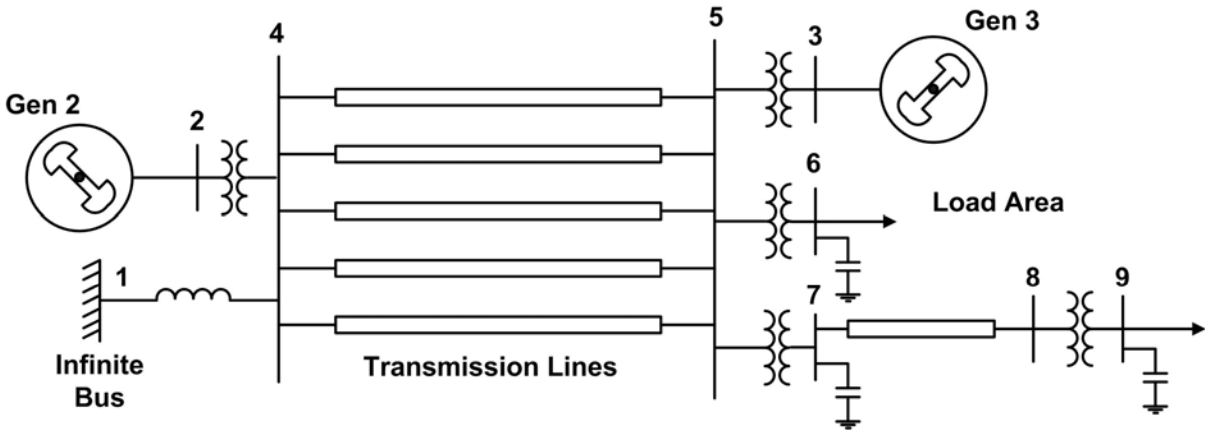

Figure B.1. Schematic diagram of the 9-bus multimachine power system.

Table B.1. 9-bus power system synchronous generator data.

Machine Parameters	Generator 2	Generator 3
Generator base power (three phase)	2200 MVA	1600 MVA
Generator line voltage (rms value)	13.2 kV	13.2 kV
Inertia	3.32 kW.s/kVA	2.32 kW.s/kVA
Base frequency	60 Hz	60 Hz
Armature time constant (T_a)	0.14 s	0.14 s
Potier Reactance	0.155 p.u	0.155 p.u
D-axis unsaturated reactance (X_d)	2.110 p.u	2.110 p.u
D-axis unsaturated transient reactance (X'_d)	0.280 p.u	0.280 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	4.2 s	4.2 s
D-axis unsaturated sub-transient reactance (X''_d)	0.215 p.u	0.215 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.032 s	0.032 s

Table B.1. (Continued)

Q-axis unsaturated reactance (X_q)	2.020 p.u	2.020 p.u
Q-axis unsaturated sub-transient reactance (X_q'')	0.215 p.u	0.215 p.u
Q-axis unsaturated sub-transient open circuit time constant (T_{qo}'')	0.062 s	0.062 s

B.2.2. Generator Automatic Voltage Regulator (AVR) and Exciter

Figure B.2 shows the schematic diagram of the AVR and exciter models used for the two generators (Venayagamoorthy *et al.*, 2003). The controllers on both synchronous machines have the same gains and time constants (Table B.2).

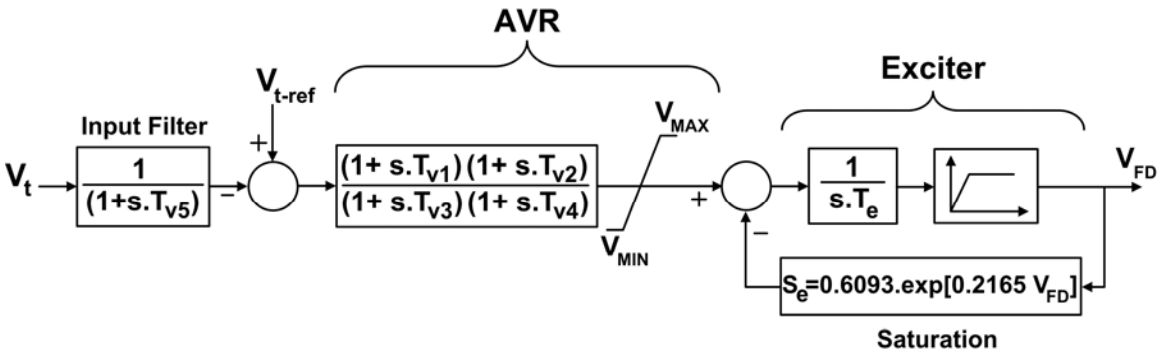

Figure B.2. Schematic diagram of the generator AVR and exciter.

Table B.2. 9-bus power system AVR and exciter data.

Parameter	Actual Values	Parameter	Actual Values
T_{v1}	0.616 s	T_{v5}	0.0235 s
T_{v2}	2.266 s	$V_{\max}$	19.59
T_{v3}	0.189 s	$V_{\min}$	-14.51
T_{v4}	0.039 s	T_e	0.47 s

B.2.3. Speed Governor and Turbine

Figure B.3 illustrates the block diagram of the speed governor and the turbine. The corresponding gains and time constants are provided in Table B.3 (Venayagamoorthy *et al.*, 2003).

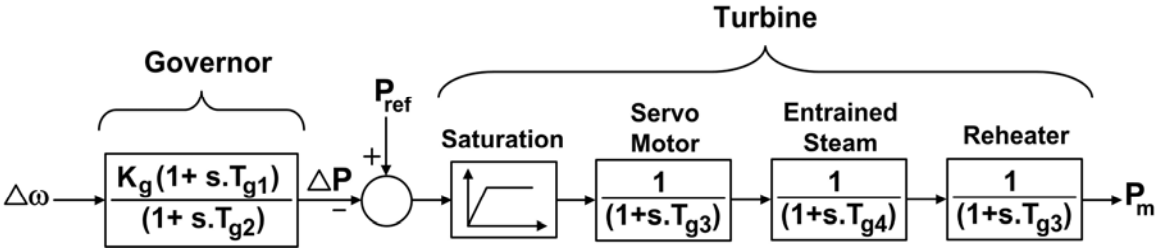

Figure B.3. Schematic diagram of the generator speed governor and turbine.

Table B.3. 9-bus power system speed governor and turbine data.

Parameter	Actual Values	Parameter	Actual Values
T_{g1}	0.264 s	T_{g5}	2.662 s
T_{g2}	0.0264 s	K_g	0.05
T_{g3}	0.15 s	$P_{\max}$	1.2 p.u
T_{g4}	0.594 s	F	0.322

B.2.4. Transmission Lines and Loads

The parameters of the transmission lines, transformers and the shunt loads in Figure B.1 are given in Table B.4. All per unit values in Table B.4 are based on a 100 MVA base. The shunt capacitors at buses 6, 7 and 9 are 5 MVar, 100 MVar and 5 MVar respectively.

Table B.4. 9-bus power system transmission line, transformer and load data.

From Bus	To Bus	Component	Voltage Level	Main Data	Tap
1	4	Reactor	545 kV	$X = 0.002$ p.u	N/A
2	4	Transformer	13.2:540 kV	2200 MVA $X = 0.0045$ p.u	-----
3	5	Transformer	13.2:530 kV	1600 MVA $X = 0.00625$ p.u	-----
4	5	Transmission Line	500 kV	Each line: $Z = 0.0015 + j \times 0.02880$ p.u $B/2 = 1.173$ p.u	N/A
5	6	Transformer	525:13.8 kV	5000 MVA $X = 0.003$ p.u	500-550 kV
5	7	Transformer	525:115 kV	5000 MVA $X = 0.003$ p.u	500-550 kV
7	8	Transmission Line	115 kV	$Z = 0.001 + j \times 0.003$ p.u	N/A
8	9	Transformer	115:13.8 kV	4000 MVA $X = 0.001$ p.u	103.5-126.5 kV
6		Load	13.8 kV	$S = 10 + j \times 6$ p.u/phase	N/A
9		Load	13.8 kV	$S = 10 + j \times 3$ p.u/phase	N/A

B.3. IEEE 12-Bus Benchmark Power System

Figure B.4 shows the IEEE 12-bus system, which is a benchmark power system designed for analyzing the dynamic and steady state effects of FACTS devices (Jinag *et al.*, 2006).

B.3.1. Generator Data

Magnetic saturation is ignored in the synchronous machines and the transformers. Each synchronus machine has a single damper winding in the q-axis (Jiang *et al.*, 2006). The details of the three generators' parameters are presented in Table B.5.

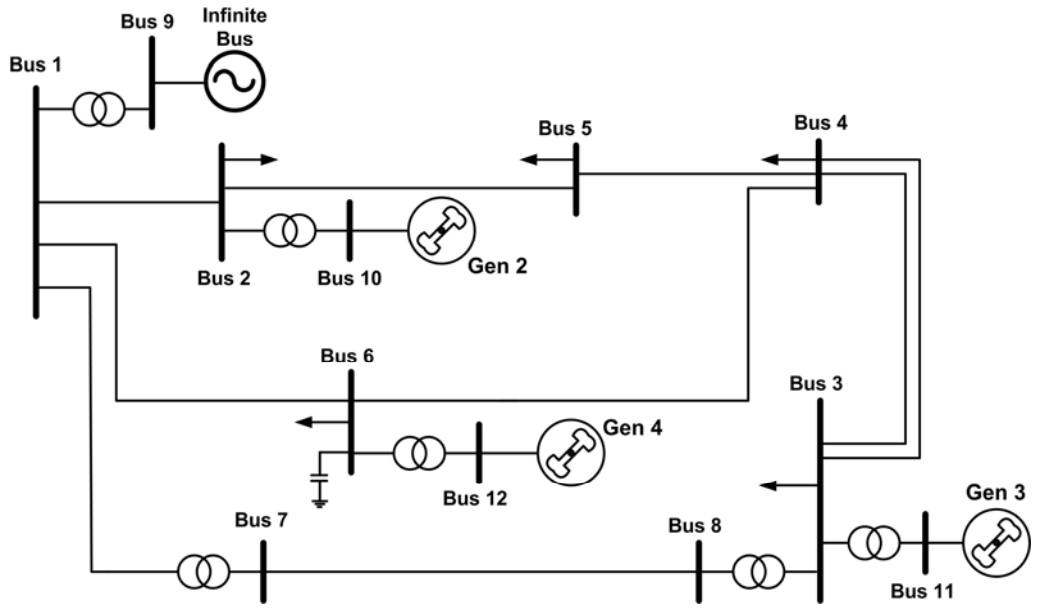

Figure B.4. Schematic diagram of the IEEE 12-bus benchmark power system.

Table B.5. IEEE 12-bus power system synchronous generator data.

Machine Parameters	Generator 2	Generator 3	Generator 4
Generator base power (three phase)	700 MVA	500 MVA	500 MVA
Generator line voltage (rms value)	22.0 kV	22.0 kV	22.0 kV
Inertia	5.0 kW.s/kVA	3.0 kW.s/kVA	5.0 kW.s/kVA
Base frequency	60 Hz	60 Hz	60 Hz
Armature resistance (R_a)	0.0002 s	0.0002 s	0.0002 s
Potier Reactance	0.012 p.u	0.01 p.u	0.012 p.u
D-axis unsaturated reactance (X_d)	1.5 p.u	1.4 p.u	1.5 p.u
D-axis unsaturated transient reactance (X'_d)	0.40 p.u	0.30 p.u	0.4 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	5.0 s	6.0 s	5.0 s
D-axis unsaturated sub-transient reactance (X''_d)	0.35 p.u	0.28 p.u	0.35 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.002 s	0.002 s	0.002 s
Q-axis unsaturated reactance (X_q)	1.2 p.u	1.35 p.u	1.2 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.35 p.u	0.27 p.u	0.35 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.002 s	0.002 s	0.002 s

All the generators are considered to have the speed governor and AVR models similar to the structures described in Sections B.2.2 and B.2.3.

B.3.2. Bus and Branch Data

The per unit values are given based on a 100 MVA base.

Table B.6. IEEE 12-bus power system bus data.

Bus	Rated Voltage (kV)	Specified Voltage (p.u)	Load (MVA)	Shunt MVar	Generation (MW)
1	230	-----	-----	-----	-----
2	230	-----	280+j×200	-----	-----
3	230	-----	320+j×240	-----	-----
4	230	-----	320+j×240	160	-----
5	230	-----	100+j×60	80	-----
6	230	-----	440+j×300	180	-----
7	345	-----	-----	-----	-----
8	345	-----	-----	-----	-----
9	22	1.05	-----	-----	-----
10	22	1.02	-----	-----	500
11	22	1.01	-----	-----	200
12	22	1.02	-----	-----	300

Table B.7. IEEE 12-bus power system branch data.

Line	Voltage (kV)	Length (km)	R (p.u)	X (p.u)	B/2 (p.u)	Rating (MVA)
1-2	230	100	0.01144	0.09111	0.091305	250
1-6	230	300	0.03356	0.26656	0.277385	250
2-5	230	300	0.03356	0.26656	0.277385	250
3-4 (1)	230	100	0.01144	0.09111	0.091305	250
3-4 (2)	230	100	0.01144	0.09111	0.091305	250
4-5	230	300	0.03356	0.26656	0.277385	250
4-6	230	300	0.03356	0.26656	0.277385	250
7-8	345	600	0.01595	0.17214	1.642650	500

Table B.8. IEEE 12-bus power system transformer data.

From	To	Voltage (kV)	Leakage Reactance (p.u)	Rating (MVA)
1	7	230:345	0.0100	1000
1	9	230:22	0.0100	1000
2	10	230:22	0.0100	1000
3	8	230:345	0.0100	1000
3	11	230:22	0.0100	1000
6	12	230:22	0.0200	500

B.4. 45-Bus 10-Generator Power System

Figure B.5 shows the 10-generator 45-bus network, which is a section of the Brazilian power system. The system has two voltage levels of 230 kV and 525 kV.

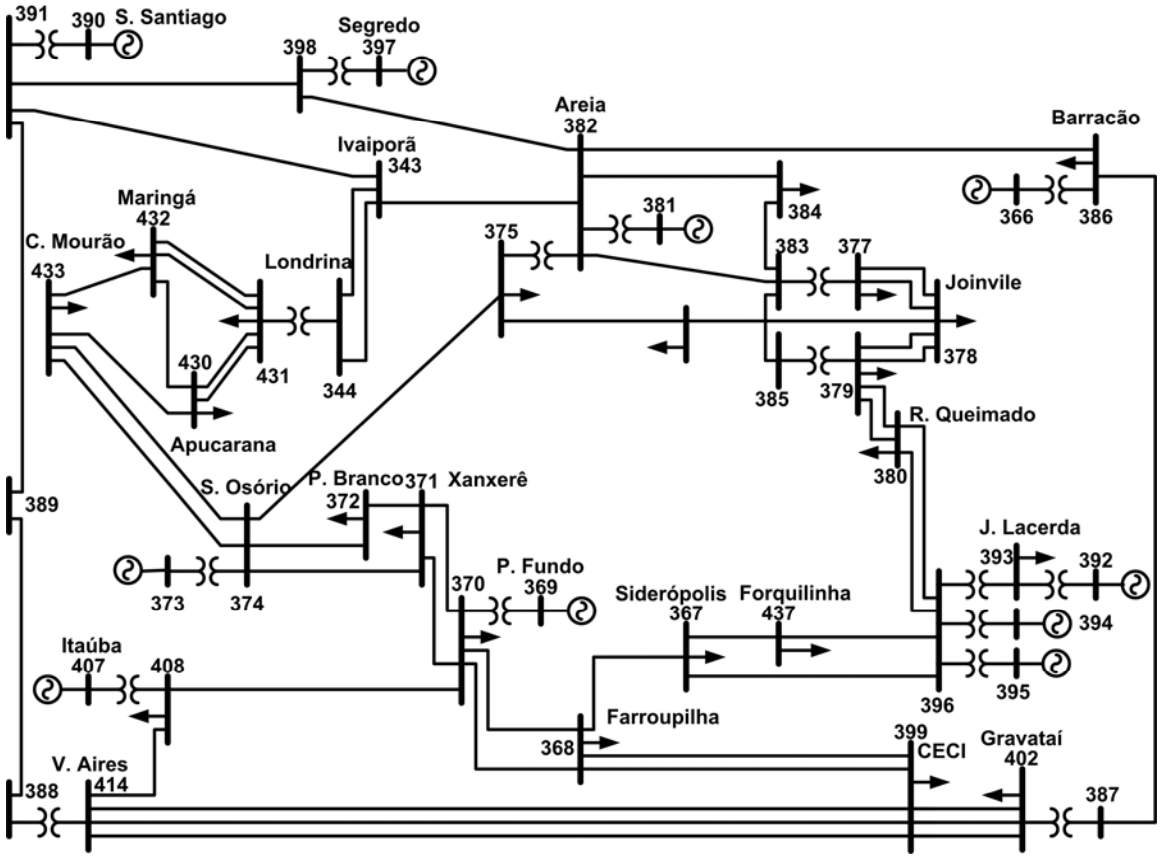

Figure B.5. Schematic diagram of the Brazilian 45-bus power system.

B.4.1. Generator Data

Many of the generator buses have more than one unit –all identical- connected to them. All the per unit values for each generator are based on the corresponding machine MVA basis itself.

Table B.9. Generator Barracao (bus 366 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	465 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	4.0 kW.s/kVA
Base frequency	60 Hz
Number of units	4
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	1.20 p.u
D-axis unsaturated transient reactance (X'_d)	0.336 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	7.92 s
D-axis unsaturated sub-transient reactance (X''_d)	0.269 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.06 s
Q-axis unsaturated reactance (X_q)	0.762 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.269 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.09 s

Table B.10. Generator P. Fundo (bus 369 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	130 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	4.349 kW.s/kVA
Base frequency	60 Hz
Number of units	2
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	0.88 p.u
D-axis unsaturated transient reactance (X'_d)	0.256 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	4.91 s
D-axis unsaturated sub-transient reactance (X''_d)	0.196 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.03 s
Q-axis unsaturated reactance (X_q)	0.622 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.196 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.04 s

Table B.11. Generator S. Osorio (bus 373 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	170 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	3.84 kW.s/kVA
Base frequency	60 Hz
Number of units	6
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	0.97 p.u
D-axis unsaturated transient reactance (X'_d)	0.32 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	8.98 s
D-axis unsaturated sub-transient reactance (X''_d)	0.27 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.035 s
Q-axis unsaturated reactance (X_q)	0.635 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.27 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.075 s

Table B.12. Generator Areia (bus 381 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	465 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	4.0 kW.s/kVA
Base frequency	60 Hz
Number of units	4
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	1.02 p.u
D-axis unsaturated transient reactance (X'_d)	0.336 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	7.92 s
D-axis unsaturated sub-transient reactance (X''_d)	0.269 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.060 s
Q-axis unsaturated reactance (X_q)	0.762 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.269 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.090 s

Table B.13. Generator S. Santiago (bus 390 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	350 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	4.086 kW.s/kVA
Base frequency	60 Hz
Number of units	4
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	0.87 p.u
D-axis unsaturated transient reactance (X'_d)	0.30 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	10.0 s
D-axis unsaturated sub-transient reactance (X''_d)	0.22 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.040 s
Q-axis unsaturated reactance (X_q)	0.59 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.22 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.130 s

Table B.14. Generator J. Lacerda A. (bus 392 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	55.5 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	3.207 kW.s/kVA
Base frequency	60 Hz
Number of units	2
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.08 p.u
D-axis unsaturated reactance (X_d)	1.47 p.u
D-axis unsaturated transient reactance (X'_d)	0.15 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	5.0 s
D-axis unsaturated sub-transient reactance (X''_d)	0.899 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.030 s
Q-axis unsaturated reactance (X_q)	1.47 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.899 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.07 s

Table B.15. Generator J. Lacerda B. (bus 394 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	156.3 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	3.097 kW.s/kVA
Base frequency	60 Hz
Number of units	1
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	2.319 p.u
D-axis unsaturated transient reactance (X'_d)	0.315 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	4.89 s
D-axis unsaturated sub-transient reactance (X''_d)	0.241 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.030 s
Q-axis unsaturated reactance (X_q)	2.319 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.241 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.056 s

Table B.16. Generator J. Lacerda C. (bus 395 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	156.3 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	3.097 kW.s/kVA
Base frequency	60 Hz
Number of units	2
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	2.319 p.u
D-axis unsaturated transient reactance (X'_d)	0.315 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	4.89 s
D-axis unsaturated sub-transient reactance (X''_d)	0.241 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.030 s
Q-axis unsaturated reactance (X_q)	2.319 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.241 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.056 s

Table B.17. Generator Segredo (bus 397 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	350.5 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	4.086 kW.s/kVA
Base frequency	60 Hz
Number of units	4
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	0.87 p.u
D-axis unsaturated transient reactance (X'_d)	0.30 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	10.0 s
D-axis unsaturated sub-transient reactance (X''_d)	0.22 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.040 s
Q-axis unsaturated reactance (X_q)	0.59 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.22 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.13 s

Table B.18. Generator Itauba (bus 407 in Figure B.5) data.

Machine Parameters	Actual Value
Generator base power (three phase)	139 MVA
Generator line voltage (rms value)	13.8 kV
Inertia	3.659 kW.s/kVA
Base frequency	60 Hz
Number of units	4
Armature resistance (R_a)	0.001 p.u
Potier Reactance	0.13 p.u
D-axis unsaturated reactance (X_d)	1.171 p.u
D-axis unsaturated transient reactance (X'_d)	0.306 p.u
D-axis unsaturated transient open circuit time constant (T'_{do})	9.55 s
D-axis unsaturated sub-transient reactance (X''_d)	0.242 p.u
D-axis unsaturated sub-transient open circuit time constant (T''_{do})	0.095 s
Q-axis unsaturated reactance (X_q)	0.81 p.u
Q-axis unsaturated sub-transient reactance (X''_q)	0.242 p.u
Q-axis unsaturated sub-transient open circuit time constant (T''_{qo})	0.125 s

B.4.2. Generator Automatic Voltage Regulator (AVR) and Exciter

A PSCAD/EMTDC® built-in model (Exciter AC1A) is used for the generator AVR and exciter. It is the standard IEEE Alternator Supplied Rectifier Excitation System #1 (Manitoba HVDC Research Center Inc., 2004) as shown in Figure B.6.

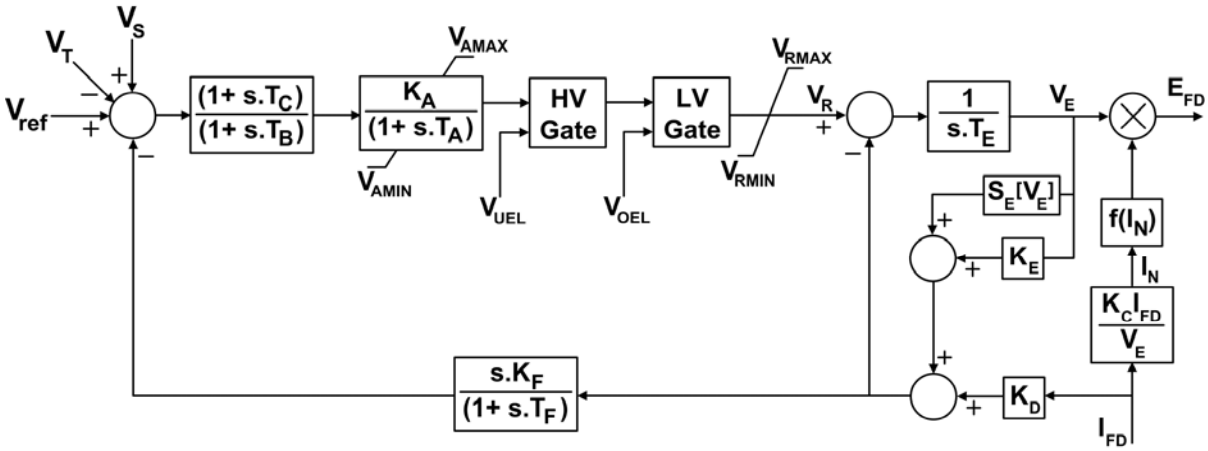

Figure B.6. Schematic diagram of the AC1A exciter in PSCAD/EMTDC®.

where:

V_T : Output of terminal voltage transducer and load compensation elements (p.u),

V_{ref} : Voltage regulator reference (p.u),

V_S : Power system stabilizer output (p.u),

V_R : Voltage regulator output (p.u),

E_{FD} : Exciter output voltage (p.u),

I_{FD} : Synchronous machine field current (p.u),

S_E : Exciter saturation function.

The values of the parameters used are the default quantities in PSCAD/EMTDC®, which are given in Table B.19.

Table B.19. Exciter AC1A (in PSCAD/EMTDC®) data.

Exciter Parameters	Actual Value
Maximum and minimum regulator output limits (V_{AMAX} , V_{AMIN})	±14.5 p.u
Voltage regulator gain (K_A)	400 p.u
Voltage regulator time constant (T_A)	0.02 s
Voltage regulator time constant (T_B)	0 s
Voltage regulator time constant (T_C)	0 s
Over-excitation and under-excitation limiters outputs (V_{OEL} , V_{UEL})	N/A
Maximum and minimum regulator output limits (V_{RMAX} , V_{RMIN})	6.03,-5.43 p.u
Exciter time constant (T_E)	0.8 s
Exciter constant related to self-excited field (K_E)	1 p.u
Demagnetizing factor (K_D)	0.38 p.u
Excitation control system stabilizer gain (K_F)	0.03 p.u
Exciter saturation function value at the corresponding exciter voltage V_E , i.e., $S_E(V_E)$	(0.1,4.18) (0.03, 3.14)
Rectifier loading factor proportional to commutation reactance (K_C)	0.2 p.u

B.4.3. Generator Speed Governor Model

A PSCAD/EMTDC® built-in component is used for modeling the generator speed governor: "Gov 1: Mechanical Hydraulic Control", whose block diagram appears in Figure B.7 (Manitoba HVDC Research Center Inc., 2004). The corresponding default parameters used are provided in Table B.20.

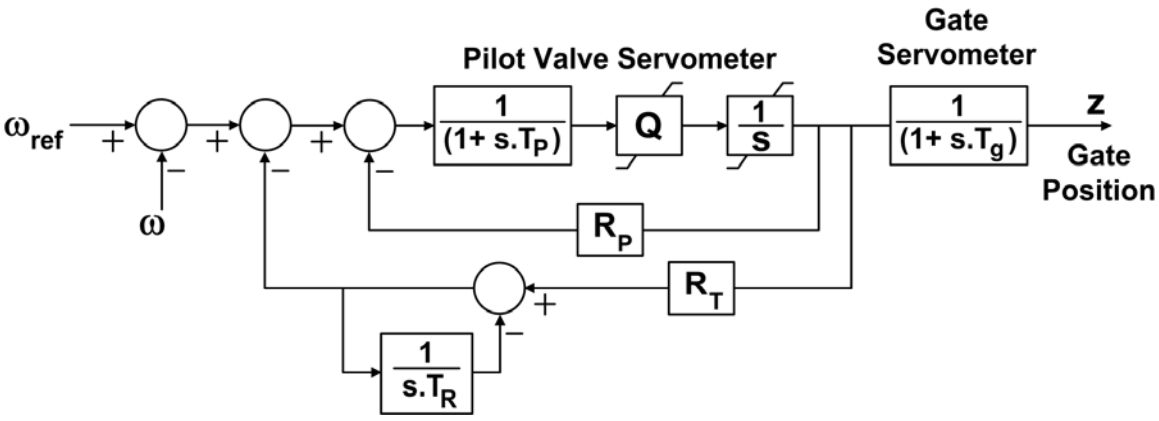

Figure B.7. Schematic diagram of the Gov1 governor in PSCAD/EMTDC®.

Table B.20. Governor Gov 1 (in PSCAD/EMTDC®) data.

Governor Parameters	Actual Value
Servo gain (Q)	5.0 p.u
Permanent droop (R_P)	0.04 p.u
Temporary droop (R_T)	0.4 p.u
Main servo time constant (T_g)	0.2 s
Pilot valve and servo motor time constants (T_p)	0.05 s
Reset or dashpot time constants (T_R)	5.0 s

B.4.4. Transmission Line, Transformer and Load Data

The transmission lines are modeled using the π-model. Table B.21 summarizes the transmission line datB. Also, Tables B.22 and B.23 provide the data on the transformers and the shunt loads/compensations. All per unit values in this section are based on a 100 MVA basis.

Table B.21. 45-bus power system transmission line data.

From Bus	To Bus	Voltage Level (kV)	R (p.u)	X (p.u)	B/2 (p.u)
391	398	525	0.0005	0.006973	0.209880
343	391	525	0.0014	0.020371	0.611978
389	391	525	0.0014	0.019551	0.598990
388	389	525	0.0022	0.029942	0.957449
382	398	525	0.0005	0.006973	0.209880
343	344	525	0.0007	0.014492	0.415085
343	344	525	0.0007	0.014492	0.415085
343	382	525	0.0018	0.022696	0.567820
382	386	525	0.0014	0.019551	0.598990
382	384	525	0.0019	0.027344	0.821858
382	383	525	0.0019	0.028028	0.839521
383	384	525	0.0005	0.006836	0.205204
383	385	525	0.0012	0.0175	0.524181
386	387	525	0.0021	0.030899	0.929396
367	368	230	0.038599	0.198728	0.084958
367	396	230	0.009599	0.049148	0.021040
367	437	230	0.003301	0.016383	0.071496
368	370	230	0.046301	0.237904	0.102109
368	370	230	0.046301	0.237904	0.102109
368	399	230	0.017699	0.091173	0.039587
368	399	230	0.017699	0.091173	0.039587
370	371	230	0.016301	0.083338	0.035997

370	371	230	0.016301	0.083338	0.035997
370	408	230	0.025	0.154566	0.117266
371	372	230	0.016301	0.083338	0.035997
371	374	230	0.031599	0.161689	0.069602
372	374	230	0.015301	0.086187	0.033604
374	375	230	0.030599	0.152429	0.067508
374	433	230	0.034401	0.175935	0.075983
374	433	230	0.034401	0.175935	0.075983
375	376	230	0.024501	0.125363	0.051054
376	377	230	0.0088	0.041313	0.130229
377	378	230	0.0182	0.09331	0.039886
377	378	230	0.0182	0.09331	0.039886
378	379	230	0.015401	0.077639	0.033704
378	379	230	0.015401	0.077639	0.033704
379	380	230	0.021599	0.110405	0.046567
379	380	230	0.021599	0.110405	0.046567
380	396	230	0.018	0.091885	0.038789
380	396	230	0.018	0.091885	0.038789
396	437	230	0.0129	0.06553	0.028219
399	402	230	0.0022	0.011397	0.005783
399	402	230	0.0022	0.011397	0.005783
399	402	230	0.0019	0.009972	0.005085
399	414	230	0.020699	0.09331	0.042977
399	414	230	0.0168	0.092597	0.042977
399	414	230	0.017599	0.098296	0.044972
408	414	230	0.0202	0.112541	0.051553
430	431	230	0.012501	0.064106	0.027721
430	431	230	0.0089	0.046299	0.019943
430	432	230	0.011	0.11824	0.050655
430	433	230	0.0229	0.117527	0.050655
431	432	230	0.0172	0.088324	0.035897
431	432	230	0.0172	0.088324	0.035897
432	433	230	0.0181	0.092597	0.0401857

Table B.22. 45-bus power system transformer data.

From	To	Voltage (kV)	X [p.u.]	Tap	Controlled Bus
344	431	525:230	0.0063	0.9-1.1 p.u	344
366	386	13.8:525	0.0136	-----	
369	370	13.8:370	0.0460	-----	
373	374	13.8:230	0.0114	-----	
375	382	230:525	0.0300	0.9-1.1 p.u	375
377	383	230:525	0.0062	0.9-1.1 p.u	377
379	385	230:525	0.0062	0.9-1.1 p.u	379
381	382	13.8:525	0.0067	-----	
387	402	525:230	0.0062	0.9-1.1 p.u	387
388	414	525:230	0.0062	0.9-1.1 p.u	388
390	391	13.8:525	0.0114	-----	
392	393	13.8:13.8	0.0871	-----	
393	396	13.8:230	0.0590	-----	
394	396	13.8:230	0.0701	-----	
395	396	13.8:230	0.0450	-----	
397	398	13.8:525	0.0068	-----	
407	408	13.8:230	0.0236	-----	

Table B.23. 45-bus power system load data.

Bus	Shunt MVar	Load (MW)	Load (MVar)
343	-200		
367		59.000	22.667
368		63.667	14.000
370		57.000	6.167
371		42.000	15.667
372		15.333	4.900
374		93.667	85.500
375		93.000	20.233
376		43.333	9.800
377		142.333	-8.333
378		103.333	47.000
379		141.333	30.200
380		39.000	17.700
383	-150		
384	-150	122.667	23.200
386	-100	58.000	-2.733
387	-150		
388	-150		
389	-150		
393		42.000	13.267
399		271.000	36.667
402		204.000	-151.667
408		134.667	45.000
414		131.000	-37.000
430		87.333	4.400
431		76.333	61.000
432		61.333	20.067
433		46.333	17.900
437		30.033	18.433

APPENDIX C

STATIC NEURAL NETWORKS: STRUCTURE AND TRAINING

C.1. Introduction

A neural network illustrated in Figure C.1 is an interconnected group of biological neurons and forms the fundamental block of the nervous system, specifically the brain. Neuron is a Greek word for “nerve cell”. The neuron receives nerve impulses through its dendrites. It then sends these impulses through its axon to the terminal buttons where the neurotransmitters are released to stimulate other neurons (Kartalopoulos, 1996).

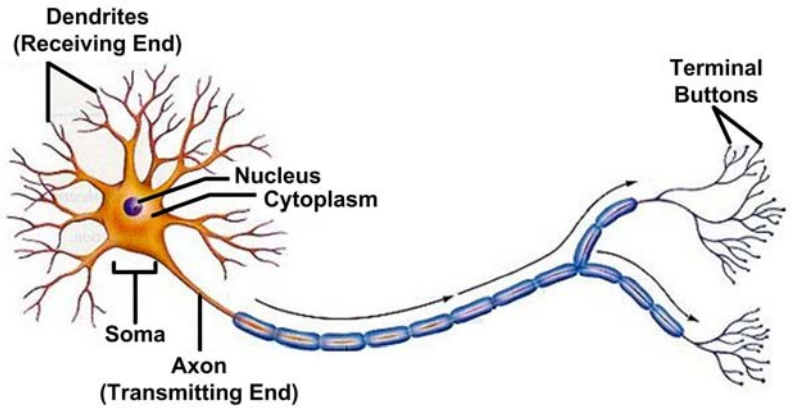

Figure C.1. Schematic diagram of a neuron.
Picture courtesy of http://www.wikipedia.org

Serious efforts have been carried out during the past 100 years to find a mathematical model for the behavior of the neurons and the nervous system. Von Neumann (1958) published one of the first papers trying to model the human brain in mathematical terms. These models are referred to as artificial neural networks, which are intelligent systems based on simplified computing models of the biological structure and functionality of the

human brain. Artificial neural networks are connectionist learning systems that are constituted of artificial neurons. An artificial neuron is a mathematical model that tries to model some properties of the biological neuron. It is a simple block with one or more inputs, a single output and a linear/nonlinear activation function. This mathematical function determines whether the artificial neuron should fire or hold in response to a specific input.

These networks have been used in the literature for a variety of applications such as system identification, function approximation, pattern recognition, control and prediction (Haykin, 1998). Artificial neural networks are classified into two categories: static neural networks and dynamic neural networks. Static networks are characterized by node equations that are memoryless; which means the node output is only a function of its current input, not of past or future inputs or outputs (Hush and Horne, 1993). Multilayer perceptron (MLP) neural networks, radial basis function (RBF) neural networks and functional link (FNL) neural networks are the most widely used static structures. Dynamic neural networks, on the other hand, are systems with memory, whose nodes are typically described by differential or difference equations (Haykin, 1998). Examples of these networks are dynamic recurrent neural networks and the Hopfield neural network.

This Appendix presents a brief overview on the most common static neural networks, as well as their structure and training process.

C.2. The Perceptron

The perceptron was introduced by Rosenblatt in 1958 as a probabilistic model for information storage and organization in the human brain (Rosenblatt, 1958). Figure C.2 shows the schematic diagram of the perceptron. The input to the perceptron is a *m*-

dimensional vector. The perceptron forms a weighted summation of the *m* entries in the input vector and passes it through the hard-limiting nonlinear function.

$$y = \Phi[\sum_{i=1}^{m} w_i \times x_i], \tag{C-1}$$

where:

$$\Phi(u) = \begin{cases} 1 & u > 0 \\ 0 & u \le 0 \end{cases}. \tag{C-2}$$

The purpose of the nonlinear function is to ensure that the neuron's response is bounded as a result of large or small activating stimuli, and therefore is controllable (Kartalopoulos, 1996). The perceptron provides a good solution for two-class pattern recognition problems where the classification problem is solved by linear separation of the input space. Moreover, simple logic functions can be effectively modeled by the perceptron (Hush and Horne, 1993).

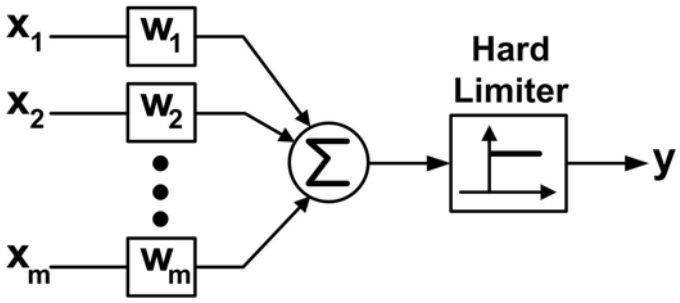

Figure C.2. Schematic diagram of the perceptron.

C.3. Multilayer Perceptron (MLP) Neural Network[1]

There are many problems that require a nonlinear partitioning of the pattern space. In addition, the perceptron cannot implement all the possible logic functions. For instance,

[1] The mathematical equations in this section are extracted and paraphrased from (Haykin, 1998).

the exclusive-or (XOR) function cannot be built by the perceptron. However, complex decision boundaries and arbitrary Boolean expression can be implemented by cascading perceptrons. The resultant network is called a multilayer perceptron (MLP) and the individual perceptrons are called neurons or nodes (Haykin, 1998; Hush and Horne, 1993).

Figure C.3 illustrates the schematic diagram of a MLP neural network which consists of three layers: an input layer, a hidden layer and an output layer. The numbers of neurons in the three layers are denoted by *m*, *n* and *r* respectively. It has been shown in the literature that a three layer MLP is capable of implementing any arbitrary logic function or forming an arbitrarily close approximation to any nonlinear decision boundary or any continuous nonlinear mapping. However, for certain problems the number of nodes in the MLP can be drastically reduced by using two hidden layers rather than one (Hush and Horne, 1993).

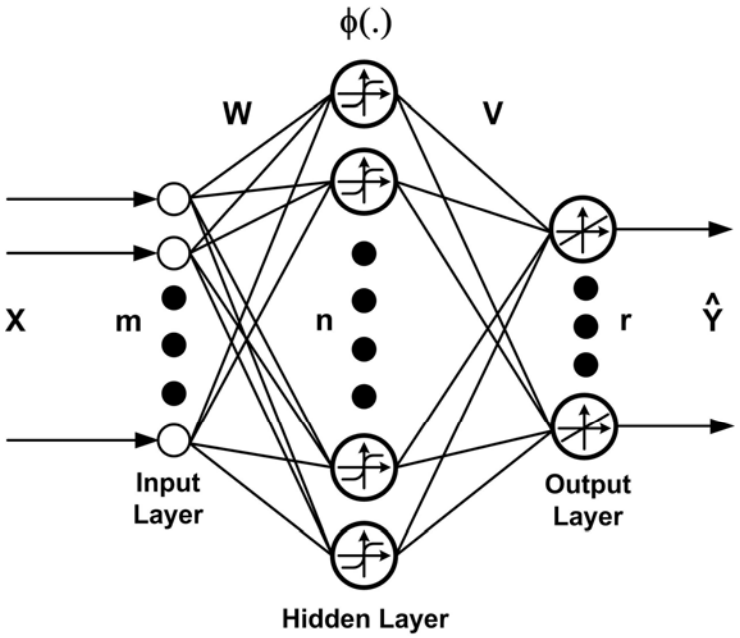

Figure C.3. Schematic diagram of the MLP neural network.

C.3.1. Feedforward Path

The feedforward path of a MLP represents the path from the input layer to the output layer. Assuming the network has *m* neurons in the input layer, the activation signal at the input of each neuron in the hidden layer can be expressed as:

$$a_j(t) = \sum_{i=1}^{m} w_{ji}(t) \times x_i(t) . \tag{C-3}$$

The activation signal of each hidden layer neuron determines its firing strength. Based on this firing strength, the output of the j^{th} neuron, referred to here as a decision signal, is given by:

$$d_j(t) = \Phi[a_j(t)] . \tag{C-4}$$

Normally in the MLP neural network, a differentiable function replaces the hard-limiting function in Figure C.2. The nonlinear activation function of the hidden layer neurons should be a continuous smooth monotonically increasing function that is bounded above and below. The most common examples are the sigmoid function:

$$d_j(t) = \frac{1}{1 + \exp[-a_j(t)]} , \tag{C-5}$$

or the hyperbolic tangent (Kartalopoulos, 1996):

$$d_j(t) = \tanh[a_j(t)] = \frac{1 - \exp[-a_j(t)]}{1 + \exp[-a_j(t)]} . \tag{C-6}$$

A weighted summation of the decision signals is now fed into each neuron in the output layer. It is quite common to use a linear function for the output nodes, which tends to make the learning process easier (Hush and Horne, 1993). The output of the k^{th} output neuron can therefore be written as:

$$\hat{y}_k(t) = \sum_{j=1}^{n} v_{kj}(t) \times d_j(t) . \tag{C-7}$$

A MLP is essentially a nonlinear mapping from an input space R^m to the output space R^r.

C.3.2. Training the MLP Network

The backpropagation algorithm is a popular choice for training the MLP neural network. This technique is based on the steepest descent method and the least-mean-square (LMS) algorithm. This section provides a mathematical background on the backpropagation algorithm and the related equations.

C.3.2.1. Unconstrained Optimization Problem[2]

The training procedure of a MLP neural network stems from the unconstrained optimization problem. Therefore, it is briefly discussed in this section:

Problem Statement: Minimize the continuously differentiable cost function $E(w)$ with respect to the weight vector w.

Iterative descent is a numerical solution to the above problem. In this method starting from an initial guess $w(0)$, a sequence of weight vectors is generated such that the cost function is reduced at each iteration of the algorithm:

$$E[w(t+1)] < E[w(t)]. \qquad \text{(C-8)}$$

Depending on the problem and the cost function, the algorithm may eventually converge to the optimal solution w^*. Three unconstrained optimization methods exist that rely on the idea of iterative descent: steepest descent method, Newton's method and Gauss-Newton method. The former is the technique used as the foundation of training MLP neural networks.

[2] The definitions and the mathematical equations in sections C.3.2.1 and C.3.2.2 are extracted from (Haykin, 1998).

C.3.2.2. Steepest Descent Method

In this method, the successive adjustments applied to the weight vector w are in the direction of the steepest descent, i.e., in the direction opposite to the gradient vector $\nabla E(w)$:

$$w(t+1) = w(t) - \eta \times \nabla E(w). \tag{C-9}$$

where η is the learning rate parameter, which in numerical computations is known as the step size. Using a first-order Taylor series expansion around $w(t)$ results in:

$$E[w(t+1)] \cong E[w(t)] + [\nabla E(w)]^T . \Delta w(t). \tag{C-10}$$

Equation (C-10) can be simplified as:

$$E[w(t+1)] \cong E[w(t)] + [\nabla E(w)]^T . (-\eta . \nabla E(w)). \tag{C-11}$$

Therefore:

$$E[w(t+1)] \cong E[w(t)] - \eta \times \| \nabla E(w) \|^2. \tag{C-12}$$

This method slowly converges to the optimal solution w^*. The selection of the learning rate parameter significantly affects the convergence of the training algorithm. Details are provided in (Haykin, 1998; Hush and Horne, 1993).

C.3.2.3. Least-Mean-Square (LMS) Algorithm

The LMS algorithm is based on the use of instantaneous values for the cost function. In this algorithm, the cost function is defined as:

$$E[w(t)] = \frac{1}{2} \| e(t) \|^2, \tag{C-13}$$

where $e(t)$ is the instantaneous error signal. The gradient vector can be simplified as:

$$\nabla E[w(t)] = \frac{\partial E[w(t)]}{\partial w(t)} = e(t) \times \frac{\partial e(t)}{\partial w(t)}. \tag{C-14}$$

Therefore, Equation (C-9) can be expressed as:

$$w(t+1) = w(t) - \eta \times e(t) \times \frac{\partial e(t)}{\partial w(t)}. \tag{C-15}$$

C.3.2.4. Backpropagation Algorithm

This section presents the general equations of the backpropagation algorithm for the MLP neural network in Figure C.3.

In the supervised learning scheme, the desired responses of the output neurons are known. Knowing the desired response $\hat{y}$ and the actual response y of the k^{th} output neuron, its corresponding instantaneous error signal can be defined as:

$$e_k(t) = y_k(t) - \hat{y}_k(t). \tag{C-16}$$

Using the LMS algorithm:

$$E(t) = \frac{1}{2}\sum_{k=1}^{r} e_k^2(t). \tag{C-17}$$

Therefore:

$$v_{kj}(t+1) = v_{kj}(t) - \eta \times e_k(t) \times \frac{\partial e_k(t)}{\partial v_{kj}(t)}. \tag{C-18}$$

Applying the chain rule gives:

$$v_{kj}(t+1) = v_{kj}(t) - \eta \times e_k(t) \times \frac{\partial e_k(t)}{\partial \hat{y}_k(t)} \times \frac{\partial \hat{y}_k(t)}{\partial v_{kj}(t)}. \tag{C-19}$$

Applying Equations (C-7) and (C-16) yields:

$$v_{kj}(t+1) = v_{kj}(t) + \eta \times e_k(t) \times d_j(t). \tag{C-20}$$

The desired responses of the hidden layer neurons are not known beforehand; therefore, the corresponding error signal for each hidden neuron has to be determined recursively in terms of the error signals of all the neurons to which that specific hidden neuron is connected. Using the basic Equation (C-9), the weight update for the input weight matrix can be written as:

$$w_{ji}(t+1) = w_{ji}(t) - \eta \times \frac{\partial E(t)}{\partial w_{ji}(t)}. \tag{C-21}$$

Using the chain rule yields:

$$w_{ji}(t+1) = w_{ji}(t) - \eta \times \frac{\partial E(t)}{\partial d_j(t)} \times \frac{\partial d_j(t)}{\partial a_j(t)} \times \frac{\partial a_j(t)}{\partial w_{ji}(t)}. \quad \text{(C-22)}$$

The first derivative term can be simplified as:

$$\frac{\partial E(t)}{\partial d_j(t)} = \frac{\partial\left[\frac{1}{2}\sum_{k=1}^{r} e_k^2(t)\right]}{\partial d_j(t)} = \frac{1}{2}\sum_{k=1}^{r}\frac{\partial e_k^2(t)}{\partial d_j(t)} = \sum_{k=1}^{r}[e_k(t) \times \frac{\partial e_k(t)}{\partial d_j(t)}]. \quad \text{(C-23)}$$

Therefore:

$$\frac{\partial E(t)}{\partial d_j(t)} = \sum_{k=1}^{r}[e_k(t) \times \frac{\partial e_k(t)}{\partial \hat{y}_k(t)} \times \frac{\partial \hat{y}_k(t)}{\partial d_j(t)}] = -\sum_{k=1}^{r}[e_k(t) \times v_{kj}(t)]. \quad \text{(C-24)}$$

The second derivative term in Equation (C-22) is the derivative of the hidden neuron activation function. If the activation function is the sigmoid function, then:

$$\frac{\partial d_j(t)}{\partial a_j(t)} = d_j(t) \times [1 - d_j(t)]. \quad \text{(C-25)}$$

For the hyperbolic tangent:

$$\frac{\partial d_j(t)}{\partial a_j(t)} = [1 + d_j(t)] \times [1 - d_j(t)]. \quad \text{(C-26)}$$

Therefore, Equation (C-21) can now be written as:

$$w_{ji}(t+1) = w_{ji}(t) + \eta \times \left[\sum_{k=1}^{r} e_k(t) \times v_{kj}(t)\right] \times \frac{\partial d_j(t)}{\partial a_j(t)} \times x_i(t). \quad \text{(C-27)}$$

A momentum term is sometimes added to the weight update equation, which helps increase the rate of learning, while avoiding the danger of instability (Haykin, 1998). For instance the entries of the input weight matrix can be written as:

$$\Delta w_{ji}(t) = \alpha \times \Delta w_{ji}(t) - \eta \times \frac{\partial E(t)}{\partial w_{ji}(t)}. \quad \text{(C-28)}$$

Equation (C-28) can now be rewritten as a time series, such that:

$$\Delta w_{ji}(t) = -\eta \times \sum_{p=0}^{t}[\alpha^{t-p} . \frac{\partial E(p)}{\partial w_{ji}(p)}], \quad \text{(C-29)}$$

Therefore, the necessary condition for convergence is that α is bound in the unit circle (Haykin, 1998). The same approach can be applied for the output weight matrix.

C.4. Radial Basis Function (RBF) Neural Network

Figure C.4 shows the schematic diagram of a radial basis function (RBF) neural network. It is a three layer network whose outputs are the linear combinations of the basis functions. The basis functions produce a localized response to input stimulus. Therefore, they produce a significant nonzero response only when the input falls within a small localized region of the input space (Hush and Horne, 1993).

In theory, the RBF network like the MLP, is capable of forming an arbitrarily close approximation to any nonlinear continuous mapping. It can be shown that the RBF is superior in classification and pattern recognition problems, while the MLP is more efficient in function approximation (Hush and Horne, 1993).

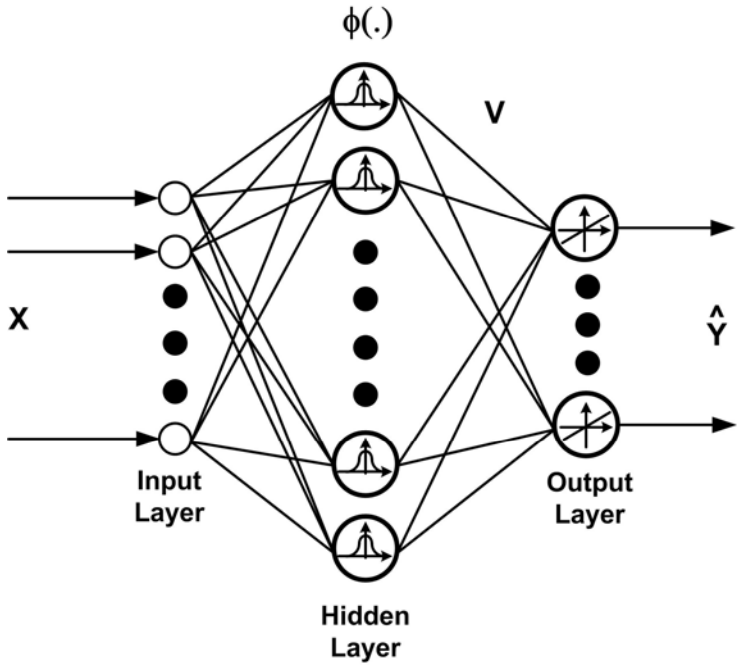

Figure C.4. Schematic diagram of the RBF neural network.

C.4.1. Feedforward Path

For the RBF neural network in Figure C.4, with the output weight links v_{kj} and the basis function $\varphi(.)$, the k^{th} output can be written as:

$$\hat{y}_k = \sum_{j=1}^{n} [v_{kj} \times \varphi\left(\frac{\| x - c_j \|_2}{\sigma_j} \right)], \tag{C-30}$$

where $x \in R^m$ is the input vector and $c_j \in R^m$ is the center of the j^{th} neuron in the hidden layer. σ_j is the width of the j^{th} center and represents a measure of the spread of the data associated with that center. The most common way is to define the center widths as the average distance between the cluster centers and the training patterns, as shown in Equation (C-31) (Hush and Horne, 1993):

$$\sigma_j = \sqrt{\frac{1}{N_j} \sum_{x \in \Gamma_j} \| x - c_j \|^2}, \tag{C-31}$$

where Γ_j represents the j^{th} cluster with center c_j and N_j number of data samples associated with it. Clearly, the center widths are calculated after the centers are assigned.

Similar to the MLP, a linear function is used for the output nodes of the RBF. Various radial basis functions have been used in the literature, such as thin-plate splines $\varphi(u) = u^2.\log(u)$, Hardy multi-quadratics $\varphi(u) = \sqrt{u^2 + c^2}$, the function $\varphi(u) = u^k$, where *k* is an odd integer and Gaussian kernel $\varphi(u) = e^{-u^2}$ (Uykan *et al.*, 2000).

C.4.2. Training the RBF Network

Training the RBF neural network consists of two stages: determining the RBF centers and adjusting the output weights. Normally, learning in the hidden layer is performed using an unsupervised method, while learning in the output layer is achieved by a supervised method (Haykin, 1998; Hush and Horne, 1993). The various learning schemes

applied to RBF neural networks differ mainly in the way they determine the centers. These schemes can be categorized as (Uykan *et al.*, 2000):

- Randomly Assigned Centers: In this approach, the centers are randomly assigned from the input training samples and remain fixed throughout the network performance,
- Orthogonalization of Regressors: These algorithms orthogonalize the regressors to decouple the contribution of each regressor to an energy based performance function,
- Supervised Selection of Centers: In this method, the centers together with the output weight matrix of the neural network are updated using a backpropagation-like gradient descent algorithm,
- Input Clustering: The locations of the centers are determined by a clustering technique applied to the input training sample vectors,
- Input-Output Clustering: In this method, clustering is applied to the augmented vectors which are obtained by concatenating the output vector to the weighted input vector and the resultant cluster vectors are then rescaled and projected onto the input space to obtain the centers.

The last two techniques are the most common approaches for RBF neural networks. These methods can be further categorized based on the way the data is fed into the neural network (Haykin, 1998):

- Batch-mode clustering of centers and batch-mode gradient descent for the linear weights,

- Batch-mode clustering of centers and pattern-mode gradient descent for the linear weights,
- Pattern-mode clustering of centers and batch-mode gradient descent for the linear weights,
- Pattern-mode clustering of centers and pattern-mode gradient descent for the linear weights.

Pattern-mode indicates that the adaptation of the parameters is done for every incoming sample and batch-mode corresponds to the case where the parameters are adapted only after collecting a number of samples.

C.4.2.1. Clustering Techniques[3]

Data clustering is an unsupervised learning problem that deals with the process of organizing different objects into groups whose members are similar in some way. The similarity of the objects can be conceptual or distance-based. A clustering algorithm can be *exclusive*, in which every object belongs to one and only one cluster or *overlapping*, where the objects can belong to two or more clusters.

The most common clustering algorithms are:

- *K-Means Clustering*: In this algorithm, a certain number of clusters are defined *a priori*. The goal is to organize the objects (data) into K clusters such that it minimizes the following objective function:

$$J = \sum_{j=1}^{K} \sum_{i=1}^{N} \| x_i^j - c_j \|^2 , \qquad \text{(C-32)}$$

[3] The material in this section is extracted and paraphrased from the tutorials on "Clustering Techniques" available at: http://fconyx.ncifcrf.gov/~lukeb and http://www.elet.polimi.it/upload/matteucc/Clustering/tutorial_html/.

where the Euclidean norm defines the distance between the i^{th} data point in the j^{th} cluster, i.e., x_i^j, and the centroid of that cluster c_j. N represents the number of the data samples.

- *Fuzzy C-Means Clustering*: This algorithm allows a piece of data to belong to two or more clusters. It is based on minimization of the following objective function:

$$J = \sum_{j=1}^{K} \sum_{i=1}^{N} u_{ij}^{m} \| x_i - c_j \|^2 , \qquad \text{(C-33)}$$

where m is any real number greater than 1, u_{ij}^{m} is the degree of membership of the i^{th} data point x_i in cluster j, with the corresponding centroid of c_j. N represents the number of the data samples.

- *Hierarchical Clustering*: Contrary to the previous techniques, in this algorithm, the number of final clusters is not known *a priori*. There are two types of hierarchical clustering:
  - Agglomerative Clustering: Each object is initially placed in its own group. The two closest groups are combined into a single group,
  - Divisive Clustering: All objects are initially placed into a single group. The two objects that are in the same group but are farthest away are used as seed points for two groups. All objects in this group are placed into the new group that has the closest seed. This procedure continues until a threshold distance is reached.
- *Probabilistic Clustering*: This is a model based approach, in which certain mathematical models are defined for the clusters and their parameters are optimized to fit the data. The models are represented by a parametric distribution,

like a Gaussian function (for continuous data) or Poisson (for discrete data). The entire data set is therefore modeled by a mixture of these distributions.

C.4.2.2. K-Means Clustering Algorithm for Assigning the RBF Centers[4]

K-Means clustering is the most widely used clustering technique and is applied throughout this study in order to determine the RBF centers. The mathematical equations and the step by step procedure for the pattern-mode and pattern-mode K-means clustering are provided in this section.

C.4.2.2.1. Pattern-Mode K-Means Clustering

Let $\{c_j(k)\}_{j=1}^{K}$ denote the centers of the RBF units at k^{th} iteration. The K-means clustering algorithm can be summarized as follows:

1. Initialization: Randomly select K points from the input data set used for finding the RBF centers. These points represent the initial centers. For the RBF neural network, K is equal to the number of neurons in the hidden layer n. Each center has the same dimension as the input to the neural network. Therefore, in the case of the multi-dimensional input space, the centers will be K vectors with the same dimension as the input vector. This set represents $\{c_j(0)\}_{j=1}^{K}$,
2. $k = k + 1$,
3. Sampling: Draw the k^{th} sample input vector $x(k) \in R^m$,
4. Metric Measurement: Compare the distance between the data sample and every individual center. The Euclidean norm is the most common metric used in this algorithm,

[4] The mathematical equations provided in this section are based on (Haykin, 1998).

5. Similarity Matching: Assign the input sample to the group that has the closest center. Find the best matching center c^* for the data sample. Let $j^*(k)$ represent the index of the best matching center c^* for the k^{th} input sample:

$$j^*(k) = \arg \min_{j=1,\ldots,K} \| x(k) - c_j(k) \|, \tag{C-34}$$

6. Center Update: Adjust the centers of the RBF units. Only the center closest to the k^{th} input sample will be updated and the other centers remain unchanged:

$$c_j(k+1) = \begin{cases} c_j(k) + \xi \times | x(k) - c_j(k) | & \text{if } j = j^*(k) \\ c_j(k) & \text{otherwise} \end{cases}, \tag{C-35}$$

where $0 < \xi << 1$ is the learning rate parameter for updating the RBF centers.

7. Repeat Steps 2-6 until the centers no longer move beyond a predefined threshold.

C.4.2.2.2. Batch-Mode K-Means Clustering

The recursive pattern-mode clustering algorithm explained above can be computationally intensive. An alternative method is to apply a batch-mode clustering algorithm to determine the RBF centers offline. As opposed to the previous method, in batch-mode training all the available data samples are presented to the RBF neural network at every epoch. Several training epochs are normally executed in order to obtain acceptable results. Once the centers are determined, their locations are kept fixed during the performance of the RBF neural network. The technique is as follows:

1. Sampling: Generate a large amount of S input samples $\{x_s \in R^m\}_{s=1}^{S}$,
2. Initialization: The n centers are initialized randomly as one of the S input samples:

$$c_j(0) = x_s, \quad j \in \{1,\ldots,n\}, \quad s \in \{1,\ldots,S\}, \tag{C-36}$$

3. Training Epoch: Go to the next training epoch $k = k+1$,

4. Similarity Matching: Assign every input sample x_s to the nearest cluster denoted by Γ_{j^*} with the center $c_j^*(k)$, by measuring a defined metric, such as the Euclidean norm of the distance between the center and the input samples,

$$\forall j \in \{1,...,n\} : \| x_s(k) - c_j^*(k) \| \leq \| x_s(k) - c_j(k) \| \Rightarrow x_s \in \Gamma_{j^*}, \quad \text{(C-37)}$$

5. Center Update: Adjust the centers of the RBF units by calculating the average position of the data samples in each cluster:

$$c_j(k+1) = \sum_{x_s \in \Gamma_j} \frac{x_s}{N_j}, \quad \text{(C-38)}$$

where N_j is the number of data samples in cluster j.

6. Stopping Criteria: Compute the error function:

$$E(k) = \sum_{j=1}^{n} \sum_{x_s \in \Gamma_j} \| x_s - c_j(k+1) \|^2, \quad \text{(C-39)}$$

If the error is lower than a predefined threshold, stop. Otherwise, repeat steps 3-5.

C.4.2.3. Backpropagation for Adjusting the Output Weights

The steepest descent based backpropagation algorithm is used for training the output weights. Similar to the equations derived in Section C.3.2.4, the output weights are adjusted according to Equation (C-40):

$$v_{kj}(t+1) = v_{kj}(t) + \eta \times e_k(t) \times d_j(t), \quad \text{(C-40)}$$

where:

$$d_j(t) = \varphi\left(\frac{\| x - c_j \|_2}{\sigma_j}\right). \quad \text{(C-41)}$$

C.5. Functional Link (FNL) Neural Network

The functional link neural network was proposed by Pao (1992), where he showed that this network can be efficiently used for function approximation and pattern recognition. Figure C.5 illustrates the schematic diagram of a FNL neural network (Patra

et al., 1999). The dimension of the input vector (*m*-dimensional) is increased by passing the input through the Function Expansion block which produces a *M*-dimensional expanded vector ($M \geq m$). The expanded vector is then passed through the output $r \times M$ weight matrix in order to generate the *r*-dimensional output vector.

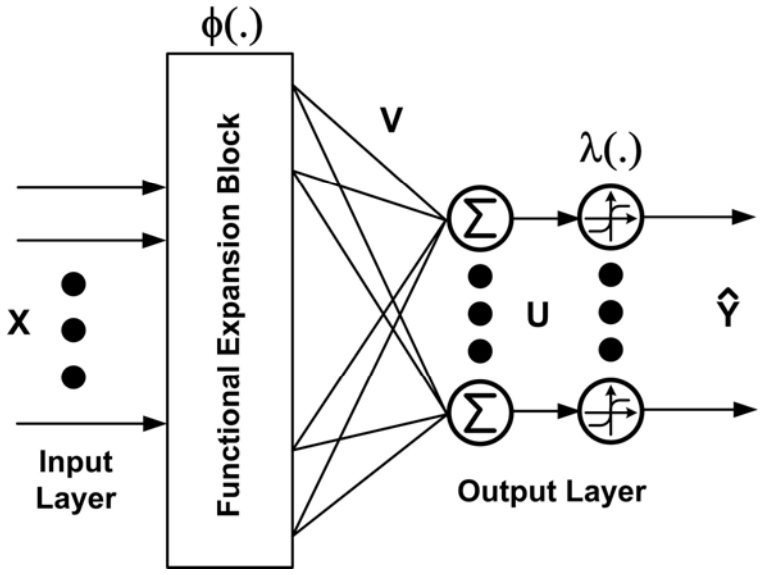

Figure C.5. Schematic diagram of the FNL neural network.

In contrast to linear weighting of the input pattern produced by the linear weights of a MLP, the functional link acts on an element or the entire pattern itself by generating a set of linearly independent functions (Patra and Kot, 2002). The functional expansion effectively increases the dimensionality of the input vector and therefore, the hyperplanes generated by the FNL provides greater discrimination capability in the input pattern space (Patra, 1999).

Although the FNL has characteristics similar to both the MLP and the RBF, it has several advantages over them. Since the hidden layer of the MLP is replaced by the

Functional Expansion block, the problems of the traditional MLP such as determination of the number of layers and processing elements is avoided (Yang and Tseng 1996). In addition, the FNL network has a faster convergence rate and less computational load than a MLP (Pao, 1992). Also, due to the removal of the hidden layer, the backpropagation algorithm becomes very simple (Patra *et al.*, 1999). Moreover, the FNL is more flexible than the MLP or the RBF since it can use a wider range of activation (basis) functions φ_j for function approximation (Sadegh, 1993). For instance, radial basis functions, splines, Boolean, sigmoid (Sadegh, 1993), Legendre polynomials (Yang and Tseng, 1996), Chebyshev polynomials (Patra and Kot, 2002) and trigonometric functions (Patra *et al.*, 1999) can all be utilized as the basis functions in a FNL. Lastly, as opposed to a RBF neural network, the FNL does not require updating the parameters of the bases.

C.5.1. Function Expansion

Yang and Tseng (1996) showed that an arbitrary function $f(x):[a,b]\rightarrow R$ can be approximated by an orthogonal function set $\{\varphi_j\}_{j=1}^{n}$:

$$F_n(x) = w_1.\varphi_1(x) + w_2.\varphi_2(x) + \cdots + w_n.\varphi_n(x)\,, \tag{C-42}$$

such that:

$$\lim_{n\rightarrow\infty} \int_a^b (f(x) - F_n(x))^2\, dx = 0\,. \tag{C-43}$$

where the weights w_j are unique and bounded. Patra *et al.* (1999) also proved that in order for the FNL neural network to converge to a set of synaptic weight solution, the bases $\{\varphi_j\}_{j=1}^{n}$ should be linearly independent. The notation $\varphi(.)$ in Figure C.5 is a general representation of the FNL bases.

Various orthogonal functions can be used for this purpose. The performance of the FNL will be more accurate by an increase in the number of the function expansion bases.

However, it should be noted that Equation (C-42) corresponds to a single dimensional function *f*. As the dimensions of the function to be approximated are increased, deriving the appropriate bases $\{\varphi_j\}$ becomes more complicated; since in many cases, the multi-dimensional orthogonal bases cannot be easily obtained.

However, several approaches exist in the literature that try to address the issue of defining bases for *m* inputs in a practical way. Some researchers have ignored the interaction between the inputs to the FNL, therefore, defining the bases as (Patra and Kot, 2002; Patra *et al.*, 1999):

$$\{\varphi_j\}_{j=1}^{M} = \{1.0, \varphi_1(x_1), \ldots, \varphi_1(x_m), \ldots, \varphi_n(x_1), \ldots, \varphi_n(x_m)\}. \quad \text{(C-44)}$$

The Functional Expansion block therefore increases the *m*-dimensional input vector to an *M*-dimensional space, where $M = m \times n + 1$. Yang and Tseng (1996) have considered the bases as the multiplication of the one dimensional orthogonal functions as in Equation (C-45):

$$\{\varphi_j\}_{j=1}^{M} = \{1.0, \varphi_1(x_1) \times \varphi_1(x_2) \times \ldots \times \varphi_1(x_m), \ldots, \varphi_n(x_1) \times \varphi_n(x_2) \times \ldots \times \varphi_n(x_m)\} \quad \text{(C-45)}$$

.

Clearly, this method increases the *m*-dimensional input vector to a (*n*+1)-dimensional space.

C.5.2. Feedforward Path

The Functional Expansion block expands the *m*-dimensional input space to a *M*-dimensional hyperspace. The outputs of the Functional Expansion block are similar to the decision signals at the output of the hidden layer neurons in a MLP (Figure C.3). These decision signals are then passed through the synaptic output weights in order to generate the signal applied to the output neurons. Unlike the common approach in the MLP neural

networks where the output nodes have a linear activation function, the output nodes of the FNL may have either linear or nonlinear activation functions. Some researchers have reported a linear activation function for the output nodes (Yang and Tseng, 1996; Sadegh, 1993; Pao *et al*., 1992), whereas others have used a hyperbolic tangent for these nodes (Patra and Kot, 2002; Patra *et al.*, 1999).

Assuming a general activation function λ is used for the output neuron activation functions, the k^{th} output of the FNL neural network can be expressed as:

$$\hat{y}_k(t) = \lambda[u_k(t)], \tag{C-46}$$

where:

$$u_k(t) = \sum_{j=1}^{M} v_{kj}(t) \times d_j(t), \tag{C-47}$$

and d_j is the j^{th} decision signal, i.e., the j^{th} output of the Function Expansion block.

C.5.3. FNL Training Process

Similar to the MLP, the FNL undergoes backpropagation for updating the synaptic output weights (Yang and Tseng, 1996). The process is considerably simpler and less computationally intensive for an FNL, due to the removal of the hidden layer.

Knowing the desired response $\hat{y}$ and the actual response y of the k^{th} output neuron, its corresponding instantaneous error signal can be defined as Equation (C-16). Applying the LMS algorithm yields:

$$v_{kj}(t+1) = v_{kj}(t) - \eta \times e_k(t) \times \frac{\partial e_k(t)}{\partial v_{kj}(t)}. \tag{C-48}$$

Applying the chain rule yields:

$$v_{kj}(t+1) = v_{kj}(t) - \eta \times e_k(t) \times \frac{\partial e_k(t)}{\partial \hat{y}_k(t)} \times \frac{\partial \hat{y}_k(t)}{\partial u_k(t)} \times \frac{\partial u_k(t)}{\partial v_{kj}(t)}, \tag{C-49}$$

which can be simplified to:

$$v_{kj}(t+1) = v_{kj}(t) + \eta \times e_k(t) \times \lambda'[u_k(t)] \times d_j(t). \tag{C-50}$$

In the special case of using the hyperbolic tangent as the output node activation function:

$$\lambda'[u_k(t)] = (1 + u_k(t)) \times (1 - u_k(t)) . \quad \text{(C-51)}$$

C.6. Time Delay (TDL) Neural Network

Static neural networks can be transformed into dynamic neural networks by converting the temporal input sequence into a static pattern by unfolding the sequence over time. In this way, time is treated as another dimension of the problem (Hush and Horne, 1993). This type of neural network was first introduced in 1988 in order to perform temporal processing (Haykin, 1998).

In order to do this, an augmented input to the static neural network is created by feeding the input sequence into a tapped delay line of finite extent. This structure is capable of modeling nonlinear systems where the output has a finite temporal dependence on the input (Hush and Horne, 1993). It provides a nonlinear autoregressive moving average (NARMA) model for the system (Narendra and Parthasarathy, 1990), and the output is given by:

$$y(k) = F[x(k), x(k-1), ..., x(k-N)] . \quad \text{(C-52)}$$

The most basic structure of the TDL neural network has no feedbacks (Figure C.6); therefore, the backpropagation algorithm can still be applied as the training algorithm. For some problems, a small feedback system is equivalent to a large and possibly infinite feedforward system. Systems with feedback are specifically efficient in identification, control and filtering applications. Neural networks can provide a solution for problems that require nonlinear feedback dynamics. This can be done by applying a feedback from the outputs of the neurons at the hidden layer, or the output layer of the neural network,

to its input sequence (Hush and Horne, 1993). This class of neural networks is referred to as Dynamic Recurrent neural network (Haykin, 1998).

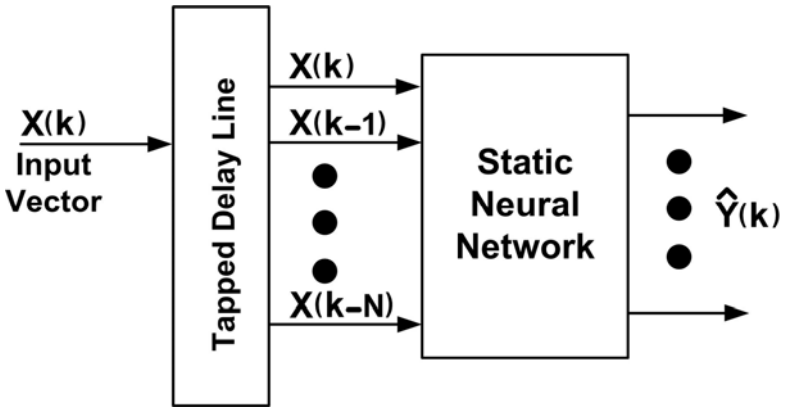

Figure C.6. Schematic diagram of the TDL neural network.

Training this type of network is more complicated since the output of the nodes is a recursive function of their output at the previous time step and therefore, the calculation of the gradient must also be a recursive computation (Hush and Horne, 1993). This class of neural network structure is beyond the scope of this study, however, a detailed explanation can be found at (Atiya and Parlos, 2000; Feldman, 1996; Haykin, 1998).

APPENDIX D

BACKPROPAGATING A SIGNAL THROUGH A NEURAL NETWORK

D.1. Introduction

The basic backpropagation equations were derived in Appendix C. These equations were related to the case where the output error is backpropagated through the neural network in order to update the synaptic weight matrices. However, in some cases a general signal needs to be backpropagated through a neural network without updating its synaptic weight matrices. Instead, the resultant backpropagated signal is used for training a second controller.

In the STATCOM indirect adaptive fuzzy controller designed in Chapter 5, it was seen that the predicted line voltage deviations at time $(t+1)$, i.e., $\Delta\hat{V}(t+1)$ should be backpropagated through the neuroidentifier in order to generate the training signal for the fuzzy controller (Figure 5.5). Also, in the case of the STATCOM action dependent ACD controller implemented in Chapters 6, 8 and 9, the training signal for the Action/fuzzy network is obtained by backpropagating a constant 1.0 through the Critic network (Figures 6.2, 8.3 and 9.4). In all these cases, the synaptic weights of the first neural network (neuroidentifier in the indirect adaptive controller or the Critic network in the ACD controller) remain unaffected.

The general mathematical equations corresponding to the case of backpropagating a signal through a MLP neural network are derived and visualized in this Appendix using a so-called B-diagram (Feldman, 1996). This approach is extremely helpful for finding the mathematical equations related to the case of backpropagating a signal through a neural

network. Clearly, the logic set forth in this approach can be readily applied in order to obtain the equations for any other type of neural network structure.

D.2. Backpropagation Diagram

Figure D.1 illustrates the schematic diagram of a MLP neural network which consists of three layers: an input layer, a hidden layer and an output layer. Without the loss of generality it is assumed that the MLP neural network shown in Figure D.1 has a single neuron in the output layer. The number of neurons in the input and hidden layers are denoted by *m* and *n* respectively.

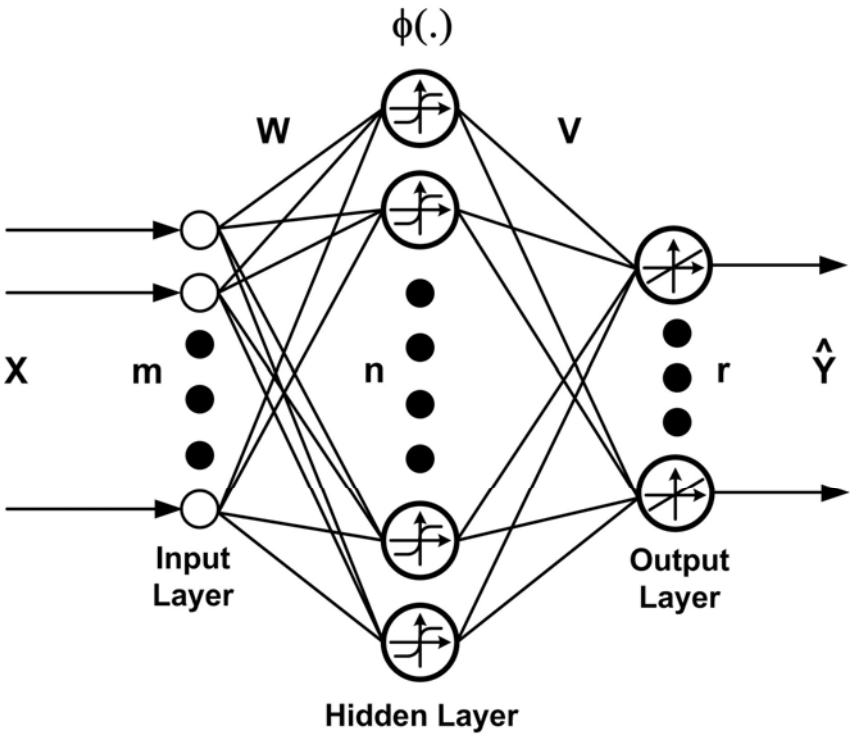

Figure D.1. Schematic diagram of the MLP neural network.

Each neuron can be represented by a composite structure (Figure D.2), where the right side computes the primitive function $f(x)$ associated with the node and left side computes the derivative of this primitive function for the same input, i.e., $f'(x)$ (Feldman 1996).

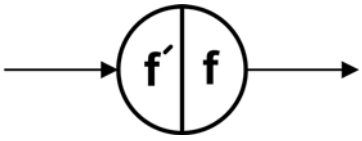

Figure D.2. The two computing sides of a neuron.

Figure D.3 illustrates two cascaded neurons with the activation functions of $f(x)$ and $g(x)$. In the feedforward (function composition) path, the output of each node is computed by applying its corresponding activation function to its input. In the backpropagation path however, the output is obtained by multiplying the input of each neuron by the value stored in its left side (Feldman, 1996).

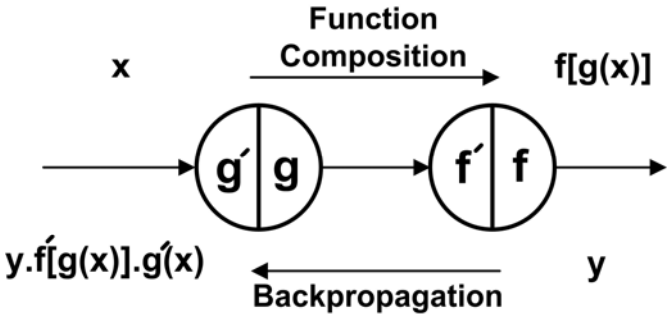

Figure D.3. Computing model of two cascaded neurons.

The backpropagation diagram is the dual of the feedforward path. Therefore, the summations and the fan-outs in the feedforward path are considered as fan-outs and summations in the backpropagation path respectively (Figure D.4). Moreover, the weighted links can be handled in the same way in both directions (Feldman, 1996). The same conventions can be used for other types of a neural network such as a RBF.

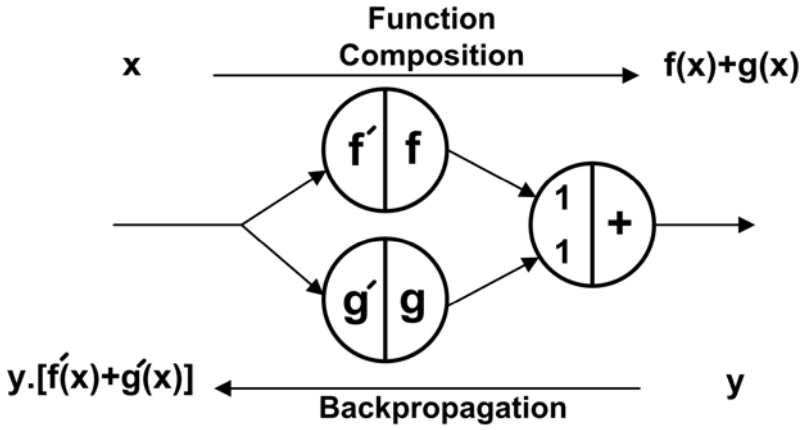

Figure D.4. Computing model of two parallel neurons.

D.3. Backpropagating a Signal Through a MLP

The objective of this section is to backpropagate a signal $u(t)$ through the neural network in Figure D.1. The equations provided here apply to the case of a MLP with a single output neuron, however, the same logic can be applied for deriving the equations for a MLP with multiple neurons in the output layer. Using the rules and definitions provided in the previous section, the backpropagation path for the j^{th} hidden neuron can be redrawn as in Figure D.5. Traditionally, linear functions are used for the input and output layers, while a nonlinear activation function is adopted for the hidden neurons.

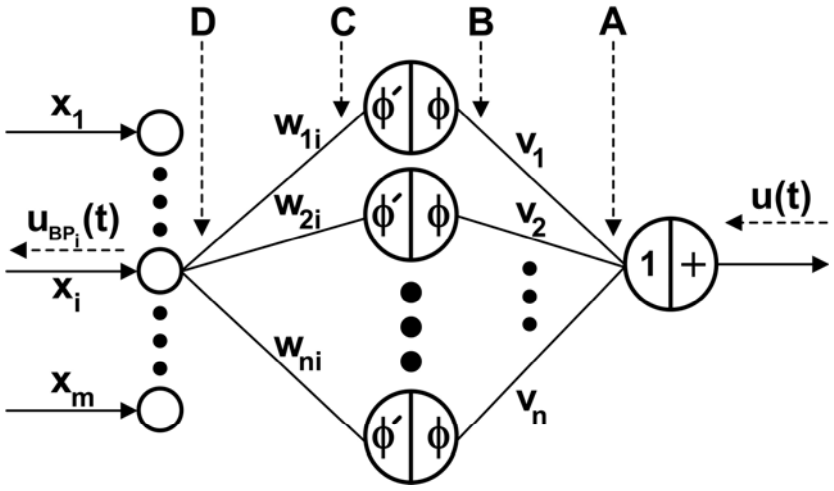

Figure D.5. Backpropagation path of a single output MLP neural network.

Since the derivative of the linear activation function at the output node is unity, the same function $u(t)$ appears at point *A* in Figure D.5.

$$u_{BP}(t)\,|_A = u(t)\,. \tag{D-1}$$

where $u_{BP}(t)\,|_A$ indicates the value of the signal $u(t)$ backpropagated through the neural network at point *A*. This signal is now multiplied by the corresponding weight link:

$$u_{BP}(t)\,|_B = u(t) \times v_j(t)\,. \tag{D-2}$$

The signal is now multiplied by the value stored in the left side of the j^{th} neuron:

$$u_{BP}(t)\,|_C = u(t) \times v_j(t) \times \frac{d\Phi[a_j(t)]}{da_j(t)}\,. \tag{D-3}$$

The signal $u_{BP}(t)\,|_C$ is now multiplied by the weight links w_{ji} in order to generate the backpropagated signal at the output of the i^{th} input neuron:

$$u_{BP_i}(t)\,|_D = u(t) \times \sum_{j=1}^{n} [v_j(t) \times \frac{d\Phi[a_j(t)]}{da_j(t)} \times w_{ji}(t)]\,. \tag{D-4}$$

Due to the fact that a llinear activation function is used for the input layer neurons, the final backpropagated signal corresponding to the i^{th} input neuron is the same as above; hence:

$$u_{BP_i}(t) = u(t) \times \sum_{j=1}^{n} [v_j(t) \times \frac{d\Phi[a_j(t)]}{da_j(t)} \times w_{ji}(t)]. \quad \text{(D-5)}$$

D.4. Input-Output Sensitivity Factor in a Neural Network

The sensitivity factor of the neural network output to its input vector is derived in this section. The special cases of a MLP and a RBF with single output nodes are considered here. However, it can be extended to any other class of neural networks with any number of neurons in the output layer.

D.4.1. Sensitivity Analysis in MLP Neural Networks

The sensitivity factor of the output of the neural network to the i^{th} input is defined as:

$$S_{x_i}^{\hat{y}} = \frac{\partial \hat{y}(t)}{\partial x_i(t)}. \quad \text{(D-6)}$$

Using the chain rule Equation (D-6) can be rewritten as:

$$\frac{\partial \hat{y}(t)}{\partial x_i(t)} = \sum_{j=1}^{n} \left[\frac{\partial \hat{y}(t)}{\partial d_j(t)} \times \frac{\partial d_j(t)}{\partial a_j(t)} \times \frac{\partial a_j(t)}{\partial x_i(t)} \right]. \quad \text{(D-7)}$$

This can in turn be simplified to:

$$\frac{\partial \hat{y}(t)}{\partial x_i(t)} = \sum_{j=1}^{n} \left[v_j(t) \times \frac{\partial \Phi_j[a_j(t)]}{\partial a_j(t)} \times w_{ji}(t) \right]. \quad \text{(D-8)}$$

Based on the type of activation functions used for the hidden neurons, their derivatives can be found by the Equations (B-25) and (B-26). Clearly, finding the sensitivity of the neural network output to the weight matrices will lead to the mathematical equation presented in Appendix C. Comparing Equations (D-5) and (D-8) yields:

$$u_{BP_i}(t) = u(t) \times \frac{\partial \hat{y}(t)}{\partial x_i(t)}. \tag{D-9}$$

D.4.2. Sensitivity Analysis in RBF Neural Networks

The same approach adopted in the previous section can be applied to a RBF neural network. However, the main differences lie in the facts that:

- a RBF has weight links of unity between the input layer and the centers,
- a RBF uses a Gaussian activation function for the neurons in the hidden layer.

A typical Gaussian function is defined as in Equation (D-10) below[1]:

$$d_j(t) = \exp\left[-\frac{\| x(t) - c_j \|^2}{\sigma_j^2}\right], \tag{D-10}$$

where $x(t)$, c_j and σ_j are the input vector and the center and the width of the j^{th} hidden neuron respectively. The derivative of the decision signal d_j with respect to the i^{th} input node can be expressed as:

$$\frac{\partial d_j(t)}{\partial x_i(t)} = \frac{\partial\left[\exp\left[-\frac{\| x(t) - c_j \|^2}{\sigma_j^2}\right]\right]}{\partial x_i(t)}, \tag{D-11}$$

which can be expanded as:

$$\frac{\partial d_j(t)}{\partial x_i(t)} = -\frac{2 \times \| x(t) - c_j \|}{\sigma_j} \times \exp\left[-\frac{\| x(t) - c_j \|^2}{\sigma_j^2}\right] \times \frac{\partial\left\{-\frac{\| x(t) - c_j \|}{\sigma_j}\right\}}{\partial x_i(t)}. \tag{D-12}$$

Also:

$$\frac{\partial\left\{\frac{\| x(t) - c_j \|}{\sigma_j}\right\}}{\partial x_i(t)} = \frac{1}{\sigma_j} \times \frac{\partial\left(\sqrt{\sum_{k=1}^{m} (x_k(t) - c_{jk})^2}\right)}{\partial x_i(t)}, \tag{D-13}$$

where c_{jk} is the k^{th} entry of the j^{th} RBF center. Therefore:

[1] Hereafter, the subscript $\| . \|_2$ denoting the 2-norm is dropped for simplicity.

$$\frac{\partial\left\{\dfrac{\| x(t)-c_j \|}{\sigma_j}\right\}}{\partial x_i(t)} = \frac{1}{\sigma_j}\times\frac{1}{2\times\| x(t)-c_j \|}\times\frac{\partial\left[\sum_{k=1}^{m}(x_k(t)-c_{jk})^2\right]}{\partial x_i(t)}. \tag{D-14}$$

And finally:

$$\frac{\partial\left\{\dfrac{\| x(t)-c_j \|}{\sigma_j}\right\}}{\partial x_i(t)} = \frac{1}{\sigma_j}\times\frac{1}{2\times\| x(t)-c_j \|}\times 2(x_i(t)-c_{ji}). \tag{D-15}$$

Hence, similar to the Equation (D-8), the sensitivity factor of the output of the RBF neural network to the i^{th} input can be expressed as:

$$\frac{\partial\hat{y}(t)}{\partial x_i(t)} = \sum_{j=1}^{n}\left[v_j(t)\times\frac{x_i(t)-c_{ji}}{\sigma_j\times\| x(t)-c_j \|}\right]. \tag{D-16}$$

D.5. Case Studies

D.5.1. Indirect Adaptive Control

In an indirect adaptive controller, the controller needs to generate a control signal that minimizes the cost function $J(t)$ defined by Equation (5-9). Therefore, the controller training signal should be (Figure 5.5):

$$e(t) = \frac{\partial J(t)}{\partial c(t)}, \tag{D-17}$$

where $c(t)$ is the controller output signal[2]. Applying the chain rule and using Equation (5-9) yields:

$$e(t) = \frac{\partial J(t)}{\partial\hat{V}(t+1)}\times\frac{\partial\hat{V}(t+1)}{\partial c(t)} = \hat{V}(t+1)\times\frac{\partial\hat{V}(t+1)}{\partial c(t)}. \tag{D-18}$$

Comparing Equation (D-18) with (D-9), and considering the fact that the signals $\hat{V}(t+1)$ and $c(t)$ are the neuroidentifier output and its 4th input (Figure 4.2), the

[2] The signal $c(t)$ is the same as the control signal $\Delta u(t)$ in Figure 4.5. However, it is been renamed here, so as not to be confused by the signal $u(t)$ used in Sections D.3 and D.4.

controller training error signal $e(t)$ can be obtained by backpropagating the RBF neuroidentifier output $\hat{V}(t+1)$ through itself.

D.5.2. Adaptive Critic Designs Based Control

In this approach, the Action network needs to generate a control signal that minimizes the cost-to-go function $J(t)$ estimated by the Critic network. Hence, the error signal applied for training the Action network should be of the form:

$$e(t) = \frac{\partial J(t)}{\partial \Delta A(t)}, \tag{D-19}$$

where $\Delta A(t)$ is the controller output signal (Figure 6.2). Considering the fact that $J(t)$ is the Critic network output and $\Delta A(t)$ is the 4^{th} input of the Critic network (Figure 6.3), following the logic in the previous section, the Action network error signal can be obtained by backpropagating the constant 1.0 through the Critic network.

APPENDIX E

MATRIX PENCIL METHOD

E.1. Introduction

This Appendix provides an overview of the matrix pencil method for approximating a function by a sum of complex exponentials[1]. The matrix pencil method is more robust to the measurement noise compared to the other linear techniques such as the polynomial method. Moreover, it has a lower variance of the estimates of the parameters of interest and is computationally more efficient (Sarkar and Pareira, 1995).

This technique is used in Chapters 5, 8 and 9 for quantitative analysis of the impact of the designed intelligent internal and supervisory level controllers on the dynamics of the power system.

E.2. Problem Formulation

In general, any measured signal can be formulated in the time domain as:

$$x(t) \approx \sum_{i=1}^{M} R_i . e^{s_i . t} \quad \text{for } 0 \le t \le T \,, \tag{E-1}$$

where R_i is the residue or the complex amplitude of the i^{th} mode s_i of the data array. For a discrete data sequence of length *N*, Equation (E-1) can be rewritten as:

$$x(k.T_s) \approx \sum_{i=1}^{M} R_i . z_i^k \quad \text{for } k = 0,..., N-1, \tag{E-2}$$

where T_s is the sampling period and the z_i's are the modes of the data sequence, which can be expressed as in Equation (E-3).

[1] Much of the material in this Appendix is extracted and paraphrased from (Sarkar *et al.*, 2000; Sarkar and Pereira, 1995).

$$z_i = e^{s_i.T_s} = e^{(-\alpha_i + j\omega_i).T_s} \text{ for } i = 1,...,M\,, \tag{E-3}$$

where α_i and ω_i denote the damping factor and the angular frequency of the i^{th} mode s_i, and M is the number of the dominant modes. The objective of the matrix pencil method is to find the best estimates of M, the R_i's and the z_i's.

One of the major advantages of the matrix pencil method over the other linear methods such as the polynomial method is that it does not have any practical limitation on the number of the dominant modes M that can be obtained by this method (Sarkar and Pereira, 1995).

E.3. Matrix Pencil Method

The assumption in this section is that the data sequence is noiseless. Nevertheless, the method can be efficiently applied to the case where the measured data is noise-contaminated (Sarkar and Pereira, 1995).

The discrete data sequence x can be expanded as in Equation (E-4):

$$x(k.T_s) = \{x(0), x(1),..., x(N-1)\}\,. \tag{E-4}$$

Two matrices $X_1, X_2 \subset R^{(N-L)\times L}$ are now constructed from the data sequence as:

$$X_1 = \begin{bmatrix} x(0) & x(1) & ... & x(L-1) \\ x(1) & x(2) & ... & x(L) \\ \vdots & \vdots & & \vdots \\ x(N-L-1) & x(N-L) & ... & x(N-2) \end{bmatrix}, \tag{E-5}$$

$$X_2 = \begin{bmatrix} x(1) & x(2) & ... & x(L) \\ x(2) & x(3) & ... & x(L+1) \\ \vdots & \vdots & & \vdots \\ x(N-L) & x(N-L+1) & ... & x(N-1) \end{bmatrix}, \tag{E-6}$$

where L is referred to as the pencil parameter. Sarkar and Pereira (1995) showed that if the pencil parameter L is chosen to be $N/2$, then the performance of the method is close to the optimal bound.

It can be shown that the two matrices can be written as (Sarkar and Pereira, 1995):

$$\begin{aligned} X_2 &= Z_1 \times R \times Z_0 \times Z_2 , \\ X_1 &= Z_1 \times R \times Z_2 , \end{aligned} \tag{E-7}$$

where $Z_1 \subset R^{(N-L)\times M}$ and $Z_2 \subset R^{M\times L}$ are defined as:

$$Z_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_M \\ \vdots & \vdots & & \vdots \\ z_1^{(N-L-1)} & z_2^{(N-L-1)} & \dots & z_M^{(N-L-1)} \end{bmatrix}, \tag{E-8}$$

$$Z_2 = \begin{bmatrix} 1 & z_1 & \dots & z_1^{L-1} \\ 1 & z_2 & \dots & z_2^{L-1} \\ \vdots & \vdots & & \vdots \\ 1 & z_M & \dots & z_M^{L-1} \end{bmatrix}, \tag{E-9}$$

and Z_0 and R are diagonal matrices in $R^{M\times M}$ defined as:

$$Z_0 = diag[z_1, z_2, ..., z_M], \tag{E-10}$$

$$R = diag[R_1, R_2, ..., R_M]. \tag{E-11}$$

A matrix pencil[2] of X_1 and X_2 can now be defined as a linear combination of the two matrices and can be expanded as in Equation (E-12):

$$X_2 - \lambda . X_1 = Z_1 \times R \times (Z_0 - \lambda . I_{M\times M}) \times Z_2 . \tag{E-12}$$

Sarkar and Pereira (1995) showed that the parameters z_i can be derived as the generalized eigenvalues of the matrix pair $\{X_1, X_2\}$. Therefore, the problem of solving

[2] If $f(t,\lambda) = g(t) + \lambda . h(t)$, then f is called a pencil of functions g and h, parameterized by λ.

for parameters z_i can be addressed as an ordinary eigenvalue problem as in Equation (E-13):

$$X_1^+ . X_2 - \lambda . I \,, \tag{E-13}$$

where X_1^+ is the Moore-Penrose pseudoinverse of X_1 and is defined as:

$$X_1^+ = \{X_1^H . X_1\}^{-1} . X_1^H \,, \tag{E-14}$$

where the superscript *H* denotes the conjugate transpose. It can be shown that the matrix product $X_1^+ . X_2$ has M nonzero eigenvalues equal to the z_i's and $(L-M)$ zero eigenvalues (Boussaïd *et al.*, 1998).

Once M and the z_i's are known, the residues R_i's can be solved by using the least squares problem in Equation (E-15):

$$\begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ z_1 & z_2 & \dots & z_M \\ \vdots & \vdots & & \vdots \\ z_1^{N-1} & z_2^{N-1} & \dots & z_M^{N-1} \end{bmatrix} \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_M \end{bmatrix}. \tag{E-15}$$

APPENDIX F

PI CONTROLLER PARAMETERS

F.1. Introduction

All the PI controllers used for the STATCOM conventional controller in this thesis are fine tuned at a single operating point. In this Appendix, the structure and the parameters of the STATCOM PI controllers as well as the operating conditions at which they are fine tuned are provided. The transfer function used for each PI controller is shown in Equation (F-1):

$$G(s) = k_P + \frac{1}{T_i s}, \tag{F-1}$$

where k_P and T_i are the proportional gain and the integral time constant of the PI controller respectively.

F.2. 9-Bus Multimachine Power System

Two PI controllers are used in the benchmark conventional controller for the STATCOM (Figure 2.11) in the 9-bus multimachine power system in Chapters 4, 5 and 6. Table F.1 provides the parameters of the two PI controllers.

Table F.1. Parameters of the STATCOM PI_V and PI_{DC} controllers in the 9-bus multimachine power system.

Controller	Proportional Gain	Integral Time Constant
PI_V	1.0	0.05
PI_{DC}	1.0	0.2

The PI_V controller used in the simulations of sections 5.5.3, 5.5.4 and 6.4.1 is equipped with an anti-windup limiter (Figure 5.14), whose parameters k_P, k_i and k_a are 1.0, 20.0 and 0.6 respectively.

The operating condition at which the PI controllers are fine tuned is given in Table F.2. The STATCOM line voltage reference and the dc link voltage reference are set at 1.01 p.u and 100 kV respectively.

Table F.2. 9-bus multimachine power system operating condition at which the STATCOM PI controllers are fine tuned.

Source	Terminal Voltage Magnitude (p.u)	Active Power (p.u)
Infinite Bus	1.0	0.39
Generator 2	1.0	0.684
Generator 3	1.01	0.686

F.3. IEEE 12-Bus Benchmark System

Two PI controllers are used in the benchmark conventional controller for the STATCOM (Figure 2.11) in the IEEE 12-bus power system in Chapters 7, 8 and 9. Table F.3 provides the parameters of the two PI controllers.

Table F.3. Parameters of the STATCOM PI_V and PI_{DC} controllers in the IEEE 12-bus power system.

Controller	Proportional Gain	Integral Time Constant
PI_V	1.0	0.054
PI_{DC}	1.0	0.4

The operating condition at which the PI controllers are fine tuned is given in Table F.4. The STATCOM line voltage reference and the dc link voltage reference are set at 1.0 p.u and 50 kV respectively.

Table F.4. IEEE 12-bus benchmark power system operating condition at which the STATCOM PI controllers are fine tuned.

Source	Terminal Voltage Magnitude (p.u)	Active Power (p.u)
Infinite Bus	1.0	0.9
Generator 2	1.02	0.7
Generator 3	1.01	0.4
Generator 4	1.02	0.68

F.4. Brazilian 45-Bus Power System

Two PI controllers are used in the benchmark conventional controller for the STATCOM (Figure 2.11) in the Brazilian 45-bus power system in Chapter 6. Table F.5 provides the parameters of the two PI controllers. The operating condition at which the PI controllers are fine tuned is given in Table F.6.

Table F.5. Parameters of the STATCOM PI_V and PI_{DC} controllers in the Brazilian 45-bus power system.

Controller	Proportional Gain	Integral Time Constant
PI_V	1.0	0.065
PI_{DC}	1.2	0.45

Table F.6. Brazilian 45-bus power system operating condition at which the STATCOM PI controllers are fine tuned.

Source	Terminal Voltage Magnitude (p.u)	Active Power (p.u)
Generator Barracao	1.02	0.97
Generator P. Fundo	1.04	0.73
Generator S. Osorio	1.01	0.96
Generator Areia	1.02	0.95
Generator S. Santiago	1.01	0.97
Generator J. Lacerda A	1.02	0.74
Generator J. Lacerda B	1.02	0.48
Generator J. Lacerda C	1.02	0.70
Generator Segredo	1.02	0.98
Generator Itauba	0.99	0.99

REFERENCES

Aboul-Ela, M.E., A.A. Sallam, J.D. McCalley, and A.A. Fouad, 1996. "Damping Controller Desing for Power System Oscillations Using Global Signals", *IEEE Transactions on Power Systems*, vol. 11, no. 2, May, pp 767-773.

Ahmad, S. and V. Tresp, 1993. "Classification with Missing and Uncertain Inputs", IEEE International Conference on Neural Networks, vol. 3, March, pp 1949-1954.

Anderson, H.C., A. Lotfi and A.C. Tsoi, 1997. "A New Approach to Adaptive Fuzzy Control: The Controller Output Error Method", *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, vol. 27, no. 4, August, pp 686-691.

Anderson, P.M. and A.A. Fouad, 1994. *Power System Control and Stability*, New York: IEEE Press, ISBN 0-7803-1029-2.

Arafeh, S.A., 1978. "Hierarchical Control of Power Distribution Systems", *IEEE Transactions on Automatic Control*, vol. AC-23, no. 2, April, pp 333-343.

Armitage, W.D. and J.C. Lo, 1994. "Enhancing the Robustness of a Feedforward Neural Network in the Presence of Missing Data", IEEE World Congress on Computational Intelligence, vol. 2, June, pp 836-839.

Åström, K. and T. Hägglund, 1995. *PID Controllers: Theory, Design and Tuning*, NC, USA: Instrumentations, Systems and Automation Society, 2nd Edition, Chapter 3, ISBN 1-55617-516-7.

Atiya, A.F. and A.G. Parlos, 2000. "New Results on Recurrent Network Training: Unifying the Algorithms and Accelerating Convergence", *IEEE Transactions on Neural Networks*, vol. 11, no. 3, May, pp 697-709.

Bahnasawi, A.A., A.S. Al-Fuhaid and M.S. Mahmoud, 1990. "Decentralised and Hierarchical Control of Interconnected Uncertain Systems", *IEE Proceedings-Part D*, vol. 137, no. 5, September, pp 311-321.

Baum, E.B. and D. Haussler, 1989. "What Size Net Gives Valid Generalization?", *Neural Computation*, no. 1, pp 151-160.

Bertsekas, D.P., 2000. *Dynamic Programming and Optimal Control*, Massachusetts: Athena Scientific, ISBN 1-886529-09-4, pp 269-359.

Boussaïd, F., F. Olivié, M. Benzohra and A. Martinez, 1998. "On the Use of the Matrix Pencil Method for Deep Level Transient Spectroscopy: MP-DLTS", *IEEE Transactions on Instrumentation and Measurement*, vol. 47, no. 3, June, pp 692-697.

Chen, C.L. and C.T. Hsieh, 1996. "User Friendly Design Method for Fuzzy Logic Controller", *IEE Proceedings- Control Theory Applications*, vol. 143, no. 4, July, pp 358-366.

Chen, S. and G. Joós, 2001. "Direct Power Control of DSTATCOMs for Voltage Flicker Mitigation", Proceedings of the 36th IEEE Industry Applications Society Annual Meeting, vol. 4, September, pp 2683-2690.

Chia, B.H.K., S. Morris and P.K. Dash, 2004. "A Fuzzy-Feedback Linearizing Nonlinear Control of CSI Based STATCOM for Synchronous Generator Stabilization", Proceedings of the IEEE International Conference on Control Applications, vol. 2, September, pp 1473-1478.

Chow, J.H., J.J. Sanchez-Gasca, H. Ren and S. Wang, 2000. "Power System Damping Controller Design Using Multiple Input Signals", *IEEE Control Systems Magazine*, August, pp 82-90.

Clarke, D.W., 2003. "Pretuning and Adaptation of PI Controllers", *IEE Proceedings-Control Theory Applications*, vol. 150, no. 6, November, pp 585-598.

Ding, Y., H. Ying and S. Shao, 2003. "Typical Takagi-Sugeno PI and PD Fuzzy Controllers: Analytical Structures and Stability Analysis", *Information Scieneces*, vol. 151, pp 245-262.

Dong, L., M.L. Crow, Z. Yang, C. Shen, L. Zhang, and S. Atcitty, 2004. "A Reconfigurable FACTS System for University Laboratories", *IEEE Transactions on Power Systems*, vol. 19, no. 1, February, pp 120-128.

Dong, L., L. Zhang and M.L. Crow, 2002. "A New Control Strategy for the Unified Power Flow Controller", Proceedings of the IEEE Power Engineering Society Winter Meeting, vol. 1, pp 562-566.

Edris, A., 2006. Private Communication, EPRI, Palo Alto, July.

El-Sharkawi, M.A. and R.J. Marks II, 2003. "Missing Data Restoration for System Control and Diagnostics", Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, Atlanta, GA, USA, August, pp 338-341.

Farsangi, M.M., Y.H. Song and Y.Z. Sun, 2000. "Supplementary Control Design of SVC and STATCOM Using H_∞ Optimal Robust Control", Proceedings of the International Conference on Electric Utility Deregulation and Restructuring and Power Technologies, April, pp 355-360.

Feldman, J., 1996. *Neural Networks: A Systematic Introduction*, Springer-Verlag, Berlin: Springer-Verlag, 1st Edition, ISBN 3-540-60505-3, Chapter 7.

Figueiredo M. and F. Gomide, 1999. "Design of Fuzzy Systems Using Neurofuzzy Networks", *IEEE Transactions on Neural Networks*, vol. 10, no. 4, July, pp 815-827.

García-González, P. and A. García-Cerrada, 2000. "Control System for a PWM-Based STATCOM", *IEEE Transactions on Power Delivery*, vol. 15, no. 4, October, pp 1252-1257.

Glover, J.D., and M. Sarma, 1986. *Power System Analysis and Design*, PWS Publishers, ISBN 0-534-07860-5, Chapter 12.

Govindhasamy, J.J., S.F. McLoone and G.W. Irwin, 2005. "Second-Order Training of Adaptive Critics for Online Process Control", *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, vol. 35, no. 2, April, pp 381-385.

Haykin, S.S., 1998. *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 2nd Edition, ISBN 0-1327-3350-1.

Han, B.M., G. Karady, J. Park and S. Moon, 1998. "Interaction Analysis Model for Transmission Static Compensator with EMTP", *IEEE Transactions on Power Delivery*, vol. 13, no. 4, October, pp 1297-1302.

Harris, J.C. and S.A. Billings, 1985. *Self-Tuning and Adaptive Control: Theory and Applications*, IEE Press, 2nd Edition, ISBN 0-8634-1036-7.

Hingorani, N.G. and L. Gyugyi, 1999. *Understanding FACTS- Concepts and Technology of Flexible AC Transmission Systems*, New Jersey: IEEE Press, ISBN 0-7803-3455-8.

Hua, Y. and T. K. Sarkar, 1990. "Matrix Pencil Method for Estimating Parameters of Exponentially Damped/Undamped Sinusoids in Noise", *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 38, May, pp. 814 – 824.

Hush, D.R. and B.G. Horne, 1993. "Progress in Supervised Neural Networks", *IEEE Signal Processing Magazine*, January, pp 8-39.

Innovative Integration, 2001. OMNIBUS User's Manual, California, USA, February.

Innovative Integration, 2000. M6x/cM6x Development Package Manual, California, USA, December.

Jang, J.S.R. and C.T. Sun, 1995, "Neuro-Fuzzy Modeling and Control", *Proceedings of the IEEE*, vol. 83, no. 3, March, pp 378-405.

Jang, J.S.R., 1993, "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 23, no. 3, May/June, pp 665-685.

Jiang, S., U.D. Annakkage and A.M. Gole, 2006. "A Platform for Validation of FACTS Models", *IEEE Transactions on Power Systems*, vol. 21, no. 1, January, pp 484-491.

Kamwa, I., R. Grondin and Y. Hébert, 2001. "Wide-Area Measurement Based Stabilizing Control of Large Power Systems- A Decentralized/Hierarchical Approach", *IEEE Transactions on Power Systems*, vol. 16, no. 1, February, pp 136-153.

Karlsson, D., M. Hemmingsson and S. Lindahl, 2004. "Wide Area System Monitoring and Control", *IEEE Power and Energy Magazine*, September/October, pp 68-76.

Kartalopoulos, S.V., 1996. *Understanding Neural Networks and Fuzzy Logic*, New York: IEEE Press, ISBN 0-7803-1128-0.

Khalil, H.K., 2002. *Nonlinear Systems*, New Jersey: Prentice Hall, 3rd Edition, ISBN 0-13-067389-7.

Kim, G.W. and K.Y. Lee, 2005. "Coordination Control of ULTC Transformer and STATCOM Based on an Artificial Neural Network", *IEEE Transactions on Power Systems*, vol. 20, no. 2, May, pp 580-586.

Klein, M., G.J. Rogers, S. Moorty and P. Kundur, 1992. "Analytical Investigation of Factors Influencing Power System Stabilizers Perfromance", *IEEE Transactions on Power Systems*, vol. EC-7, September, pp 382-388

Krajewski, W., A. Lepschy and U. Viaro, 2004. "Designing PI Controllers for Robust Stability and Performance", *IEEE Transactions on Control Systems Technology*, vol. 12, no. 6, pp 973-983.

Kristiansson, B. and B. Lennartson, 2002. "Robust and Optimal Tuning of PI and PID Controllers", *IEE Proceedings- Control Theory Applications*, vol. 149, no. 1, January, pp 17-25.

Kundur, P., 1994. *Power System Stability and Control*, McGraw-Hill, Inc., ISBN 0-0703-5958-X.

Lee, C.C., 1990. "Fuzzy Logic in Control Systems: Fuzzy Logic Controller, Parts I&II", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, no. 2, March/April, pp 404-430.

Lehn, P.W. and M.R. Iravani, 1998. "Experimental Evaluation of STATCOM Closed Loop Dynamics", *IEEE Transactions on Power Delivery*, vol. 13, no. 4, October, pp 1378-1384.

Lendaris, G.L. and J.C. Neidhoefer, 2004. "Guidance in the Use of Adaptive Critics for Control", in *Handbook of Learning and Approximate Dynamic Programming*, J. Si, A.G. Barto, W.B. Powell and D.C. Wunsch II, Piscataway, NJ, ISBN 0-471-66054-X, Chapter 4, pp 97-124.

Li, Q., Q. Jiang, Z. Wang and D. Retzmann, 1998. "Design of a Rule-Based Controller for STATCOM", Proceedings of the Annual Conference of the IEEE Industrial Electronics Society, vol. 1, September, pp 467-472.

Lin, C.T. and C.S.G. Lee, 1994. "Reinforcement Structure/Parameter Learning for Neural-Network-Based Fuzzy Logic Control Systems", *IEEE Transactions on Fuzzy Systems*, vol. 2, no. 1, February, pp 46-63.

Lin, C.T. and C.S.G. Lee, 1991. "Neural-Network-Based Fuzzy Logic Control and Decision System", *IEEE Transactions on Computers*, vol. 40, no. 12, December, pp 1320-1336.

Liu, F., S. Mei, Q. Lu, Y. Ni, F.F. Wu and A. Yokoyama, 2003. "The Nonlinear Internal Control of STATCOM: Theory and Application", *International Journal of Electrical Power & Energy Systems*, vol. 25, no. 6, pp. 421 – 430.

Lu, Q., F. Liu, S. Mei and M. Goto, 2001. "Nonlinear Disturbance Attenuation Control for STATCOM", IEEE Power Engineering Society Winter Meeting, Columbus, OH, USA, vol. 3, pp 1323-1328.

Manitoba HVDC Research Center Inc, 2004. "Introduction to PSCAD/EMTDC® Version 4.1.1".

Mathur, M. and R.K. Varma, 2002. *Thyristor-Based FACTS Controllers for Electrical Transmission Systems*, New York: Wiley- IEEE Press, ISBN 0-4712-0643-1.

Mendel, J.M., 2001. *Uncertain Rule-Based Fuzzy Logic Systems*, New Jersey: Prentice Hall, ISBN 0-13-040969-3.

Mendel, J.M. and G.C. Mouzouris, 1997. "Designing Fuzzy Logic Systems", *IEEE Transactions on Circuits and Systems II- Analog and Digital Signal Processing*, vol. 44, no. 11, November, pp 885-895.

Milano, F., 2006. *PSAT: Power Systems Analysis Toolbox*, Documentation for PSAT version 2.0.0 β_1, March, pp 1-477, available at http://www.power.uwaterloo.ca/~fmilano/.

Miller, T.J.E, 1982. *Reactive Power Control in Electric Systems*, John Wiley & Sons, ISBN 0-4718-6933-3.

Mohaddes, M., A.M. Gole and P.G. McLaren, 1999. "A Neural Network Controlled Optimal Pulse Width Modulated STATCOM", *IEEE Transactions on Power Delivery*, vol. 14, no. 2, April, pp 481-488.

Mohan, N., T.M. Undeland, and W.P. Robbins, 1995. *Power Electronics- Converters, Applications and Design*, 2nd ed., New York: John Wiley & Sons, Inc., ISBN 0-471-58408-8, Chapter 6.

Mohagheghi, S., Y. del Valle, G.K. Venayagamoorthy and R.G. Harley, 2006a. "A Proportional-Integrator Type Adaptive Critic Design Based Neurocontroller for a Static Compensator in a Multimachine Power System", To Appear in the *IEEE Transactions on Industrial Electronics*.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2006b. "Neural Network Based Wide Area State Predictor and Optimal Control System for the IEEE 12-Bus Benchmark System with a STATCOM", Accepted for Publication in the Proceedings of the 41st IEEE Industry Applications Society Annual Conference (IAS'06), Tampa, FL, USA, October.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2006c. "Adaptive Critic Designs Based Coupled Neurocontrollers for a Static Compensator", Accepted for Publication in the Proceedings of the International Symposium on Intelligent Control (ISIC), Munich, Germany, October.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2006d. "An Interval Type-II Robust Fuzzy Logic Controller for a Static Compensator in a Multimachine Power System", Accepted for Publication in the Proceedings of the World Congress on Computational Computations (WCCI), Vancouver, BC, Canada, July.

Mohagheghi, S., Y. del Valle, W. Qiao, G.K. Venayagamoorthy, R.G. Harley, D.M. Falcão, G.N. Taranto and T.M.L. Assis, 2006e. "Steady State and Transient Behavior of a Fuzzy-Based Controlled STATCOM Optimally Placed in a 45 Bus Brazilian Power System", Accepted for Publication in the Proceedings of the 10th Symposium of Specialists in Electric Operational and Expansion Planning (X SEPOPE), Florianopolis, Brazil, May.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2006f. "Adaptive Critic Design Based Neuro-Fuzzy Controller for a Static Compensator in a Multimachine Power System", To Appear in the *IEEE Transactions on Power Systems*.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2005a. "A Dynamic Recurrent Neural Network for Wide Area Identification of a Multimachine Power System with a FACTS Device", Proceedings of the 13th International Conference on

Intelligent Systems Application to Power Systems (ISAP), Arlington, VA, USA, November, pp 215-220.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2005b. "An Adaptive Mamdani Fuzzy Logic Based Controller for a Static Compensator in a Multimachine Power System", Proceedings of the 13th International Conference on Intelligent Systems Application to Power Systems (ISAP), Arlington, VA, USA, November, pp 228-233.

Mohagheghi, S., G.K. Venayagamoorthy, S. Rajagopalan and R.G. Harley, 2005c. "Hardware Implementation of a Mamdani Based Fuzzy Logic Controller for a Static Compensator in a Multimachine Power System", Proceedings of the 40th IEEE Industry Applications Society Annual Conference (IAS'05), Hong Kong, October, vol. 2, pp 1286-1291.

Mohagheghi, S., Y. del Valle, G.K. Venayagamoorthy and R.G. Harley, 2005d. "A Comparison of PSO and Backpropagation for Training RBF Neural Networks for Identification of a Power System with STATCOM", Proceedings of the 2005 IEEE Swarm Intelligence Symposium (SIS), Pasadena, CA, USA, June, pp 381-384.

Mohagheghi, S., W. Qiao, G.K. Venayagamoorthy and R.G. Harley, 2005e. "Supervisory Level Identifier for a Multimachine Power System Using Radial Basis Function Neural Networks", Proceedings of the 6th International Conference on Power Systems Operation and Planning (ICPSOP), Praia, Republic of Cape Verde, May, pp 134-138.

Mohagheghi, S., G.K. Venayagamoorthy and R.G. Harley, 2004a. "Modified Takagi-Sugeno Fuzzy Logic Based Controllers for a Static Compensator in a Multimachine Power System", 39th IEEE Industry Applications Society Annual Conference, Seattle, WA, USA, October, pp 2637-2642.

Mohagheghi, S., R.G. Harley and G.K. Venayagamoorthy, 2004b. "Supervisory Level Neural Network Identifier for a Small Power System with a STATCOM and a Generator", Proceedings of the IEEE-INNS International Joint Conference on Neural Networks, Budapest, Hungary, July, pp 2901-2906.

Mohagheghi, S., R.G. Harley and G.K. Venayagamoorthy, 2004c. "Intelligent Control Schemes for a Static Compensator Connected to a Power Network", Proceedings of the Second IEE International Conference on Power Electronics, Machines and Drives, Edinburgh, UK, April, vol. 2, pp 594-599.

Mohagheghi, S., R.G. Harley, G.K. Venayagamoorthy and M.L. Crow, 2003a. "Adaptive Critic Design Based Neurocontroller for a STATCOM Connected to a Power System", 38th IEEE Industry Applications Society Annual Conference, Salt Lake City, UT, USA, October, pp 749-754.

Mohagheghi, S., R.G. Harley, G.K. Venayagamoorthy and M.L. Crow, 2003b. "Indirect Adaptive Neurocontrol Scheme for a Static Compensator Connected to a Power System", International Federation of Automatic Control (IFAC) Symposium on Power Plants and Power Systems Control, Seoul, Korea, September, pp 1188-1193.

Mohagheghi, S., J.W. Park, R.G. Harley, G.K. Venayagamoorthy and M.L. Crow, 2003c. "An Adaptive Neural Network Identifier for Effective Control of a Static Compensator Connected to a Power System", Proceedings of the IEEE-INNS International Joint Conference on Neural Networks, Portland, OR, USA, July, vol. 4, pp 2964-2969.

Monticelli, A., 2000. "Electric Power System Estimation", *Proceedings of the IEEE*, vol. 88, no. 2, February, pp 262-281.

Morris, S., P.K. Dash and K.P. Basu. 2003, "A Fuzzy Variable Structure Controller for STATCOM", *International Journal of Electrical Power & Energy Systems*, vol. 25, no. 6, pp 1-12.

Narayanan, S., R.J. Markds II, J.L. Vian, M.A. El-Sharkawi and B.B. Thompson, 2002. "Set Constraint Discovery: Missing Sensor Data Restoration Using Auto-Assocoative Regression Machines", International Joint Conference on Neural Networks, May, vol. 3, pp 2872-2877.

Narendra, K.S., 1992. "Adaptive Control of Dynamical Systems Using Neural Networks", in *Handbook of Intelligent Control*, White and D.A. Sofge, Massachusetts: Van Nostrand Reinhold, ISBN 0-442-30857-4, Chapter 5, pp 141-182.

Narendra, K.S. and K. Parthasarathy, 1990. "Identification and Control of Dynamical Systems Using Neural Networks", IEEE Transactions on Neural Networks, vol. 1, no. 1, March, pp 4-27.

Ni, H., G.T. Heydt and L. Mili, 2002. "Power System Stability Agents Using Robust Wide Area Control", *IEEE Transactions on Power Systems*, vol. 17, no. 4, November, pp 1123-1131.

Okou, F.A., L.A. Dessaint and O. Akhrif, 2004. "Large Power System Stability Enhancement Using Wide-Area Signals Based Hierarchical Controller", Proceedings of the IEEE Power Engineering Society General Meeting, vol. 2, June, pp 1583-1589.

Pao, Y.H., S.M. Phillips and D.J. Sobajic, 1992. "Neural-Net Computing and Intelligent Control Systems", International Journal of Control, vol. 56, no. 2, pp 263-289.

Passino, K.M. and S. Yurkovich, 1997. *Fuzzy Control*, California: Addison-Wesley, ISBN 0-201-18074-X.

Patil, K.V., J. Senthil, J. Jiang, and R.M. Mathur, 1998, "Application of STATCOM for Damping Torsional Oscillations in Series Compensated AC Systems", *IEEE Transactions on Energy Conversion*, vol. 13, no. 3, September, pp 237-243.

Patra, J.C. and A.C. Kot, 2002. "Nonlinear Dynamic System Identification Using Chebyshev Functional Link Aritificial Neural Networks", *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, vol. 32, no. 4, August, pp 505-511.

Patra, J.C., R.N. Pal, B.N. Chatterji and G. Panda, 1999. "Identification of Nonlinear Dynamic Systems Using Functional Link Artificial Neural Networks", *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, vol. 29, no. 2, April, pp 254-262.

Pereira-Filho, O.M. and T. K. Sarkar, 1995. "Using the Matrix Pencil Method to Estimate the Parameters of a Sum of Complex Exponentials", *IEEE Transactions on Antennas and Propagation*, vol. 37, pp 48-55.

Petitclair, P., S. Bacha and J.P. Rognon, 1996. "Averaged Modeling and Nonlinear Control of an Advanced Static Var Compensator", Proceedings of the Power Electronics Specialists Conference, vol. 1, June, pp 753-758.

Poulin, É. and A. Pomerleau, 1996. "PID Tuning for Integrating and Unstable Processes", *IEE Proceedings- Control Theory Applications*, vol. 143, no. 5, September, pp 429-435.

Posser, J., J. Selinsky, H. Kwanty and M. Kaml, 1995. "Supervisory Control of Electric Power Transmission Networks", *IEEE Transactions on Power Systems*, vol. 10, no. 2, May, pp 1104-1110.

Povh, D., 2000. “Use of HVDC and FACTS”, *Proceedings of the IEEE*, vol. 88, no. 2, February, pp 235-245.

Prokhorov, D.V. and D.C. Wunsch, II, 1997. “Adaptive Critic Designs”, *IEEE Transactions on Neural Networks*, vol. 8, no. 5, September, pp 997-1007.

Qu, S. and C. Chen, 2002. “Low Frequency Damping by STATCOM with a Fuzzy Supplementary Controller”, Proceedings of the International Conference on Power System Technology, vol. 1, October, pp 67-70.

Rao, P., M.L. Crow and Z. Yang, 2000. “STATCOM Control for Power System Voltage Control Applications”, *IEEE Transactions on Power Delivery*, vol. 15, no. 4, October, pp 1311-1317.

Reed, R. 1993. “Pruning Algorithms- A Survey”, *IEEE Transactions on Neural Networks*, vol. 4, no. 5, pp 740-747.

Rosenblatt, F., 1958. “The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain”, *Psychological Review*, vol. 65, pp 386-408.

Rovlak, J.A. and R. Corlis, 1991. “Dynamic Matrix Based Control of Fossil Power Plants”, *IEEE Transactions on Energy Conversion*, vol. 6, no. 2, June, pp 320-326.

RTDS - A Fully Digital Power System Simulator Operating in Real Time, 1995. Presented at ICDS-95, College Station, TX, USA, April.

Rubaai, A. and F.E. Villaseca, 1989. “Transient Stability Hierarchical Control in Multimachine Power Systems”, *IEEE Transactions on Power Systems*, vol. 4, no. 4, October, pp 1438-1444.

Rudie, K., and W.M. Wonham, 1992. “Think Globally, Act Locally: Decentralized Supervisory Control”, *IEEE Transactions on Automatic Control*, vol. 37, no. 11, November, pp 1692-1708.

Sadegh, N., 1993. “A Perceptron Network for Functional Identification and Control of Nonlinear Systems”, *IEEE Transactions on Neural Networks*, vol. 4, no. 6, November, pp 982-988.

Sarkar, T.K., S. Park, J. Koh and S.M. Rao, 2000. "Application of the Matrix Pencil Method for Estimating the SEM (Singularity Expansion Method) Poles of Source-Free Transient Responses from Multiple Look Directions", *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 4, April, pp 612-618.

Sarkar, T.K. and O. Pereira, 1995. "Using the Matrix Pencil Method to Estimate the Parameters of a Sum of Complex Exponentials", *IEEE Antennas and Propagation Magazine*, vol. 37, no. 1, February, pp 48-55.

Schauder, C., M. Gernhardt, E. Stacey, T. Lemak, L. Gyugyi, T.W. Cease and A. Edris, 1995. "Development of a ±100 MVAR Static Condenser for Voltage Control of Transmission Systems", *IEEE Transactions on Power Delivery*, vol. 10, no. 3, July, pp 1486-1496.

Schauder, C., and H. Mehta, 1993. "Vector Analysis and Control of Advanced Static VAR Compensators", *IEE Proceedings-C*, vol. 140, no. 4, July, pp 299-306.

Schweppe, F.C. and E.J. Handschin, 1974. "Static State Estimation in Electric Power Systems", *IEEE Proceedings*, vol. 62, July, pp 972-982.

Shen, D. and P.W. Lehn, 2002. "Modeling, Analysis and Control of a Current Source Inverter-Based STATCOM", *IEEE Transactions on Power Delivery*, vol. 17, no. 1, January, pp 248-253.

Shen, D., X. Linag and H. Yingduo, 2000, "A Modified Per-Unit STATCOM Model and Analysis of Open Loop Response Time", Proceedings of the IEEE Power Engineering Winter Meeting, vol. 4, January, pp 2624-2629.

Si, J., L. Yang and D. Liu, 2004. "Direct Neural Dynamic Programming", in *Handbook of Learning and Approximate Dynamic Programming*, J. Si, A.G. Barto, W.B. Powell and D.C. Wunsch II, Piscataway, NJ, ISBN 0-471-66054-X, Chapter 5, pp 125-152.

Si, J. and Y.T. Wang, 2001. "On-Line Learning Control by Association and Reinforcement", *IEEE Transactions on Neural Networks*, vol. 12, no. 2, March, pp 264-276.

Singh, B.N., A. Chandra and K. Al-Haddad, 2000. "DSP-Based Indirect-Current-Controlled STATCOM: I. Evaluation of Current Control Techniques", *IEE Proceedings- Electric Power Applications*, vol. 147, no. 2, March, pp 107-112.

Song, Y.H. and A.T. Johns, 1998. "Application of Fuzzy Logic in Power Systems, Part 2: Comparison and Integration with Expert Systems, Neural Networks and Genetic Algorithms", *IEE Power Engineering Journal*, August, pp 185-190.

Suchomski, P., 2001. "Robust PI and PID Controller Design in Delta Domain", *IEE Proceedings- Control Theory Applications*, vol. 148, no. 5, September, pp 350-354.

Sugeno, M. and G.T. Kang, 1988. "Structure Identification of Fuzzy Model", *Fuzzy Sets and Systems*, vol. 28, pp 15-33.

Sun, J., D. Czarkowski, and Z. Zabar, 2002. "Voltage Flicker Mitigation Using PWM-Based Distribution STATCOM", Proceedings of the IEEE Power Engineering Society Summer Meeting, vol. 1, pp 616-621.

Takagi, T. and M. Sugeno, 1985. "Fuzzy Identification of Systems and Its Applications to Modeling and Control", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, pp 116-132.

Taylor, C.W., M.V. Venkatasubramanian, and Y. Chen, 2000. "Wide Area Stability and Voltage Control", Proceedings of the 7th Symposium of Specialists in Electric Operational and Expansion Planning, Curitiba, Brazil, May.

Thompson, B.B., R.J. Marks II and M.A. El-Sharkawi, 2003. "On the Contractive Nature of Autoencoders: Application to Missing Sensor Restoration", International Joint Conference on Neural Networks, Budapest, Hungary, July, vol. 4, pp 3011-3016.

Uykan, Z., C. Güzeliş, M.E. Celebi and H.N. Koivo, 2000. "Analysis of Input-Output Clustering for Determining Centers of RBFN", *IEEE Transactions on Neural Networks*, vol. 11, no. 4, July, pp 851-858.

Valderrama, G.E., P. Mattavelli, and A.M. Stanković, 2001. "Reactive Power and Unbalance Compensation Using STATCOM with Dissipativity-Based Control", *IEEE Transactions on Control Systems Technology*, vol. 9, no. 5, September, pp 718-727.

Vapnik, V.N. and A.Ya. Chervonenkis, 1971. "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities", *Theory of Probability and Its Application*, vol. 16, no. 2, pp 264-280.

Van Neumann, J., 1958. *The Computer and the Brain*. New Haven, Connecticut: Yale University Press.

Venayagamoorthy, G.K. and S. Ray, 2005. "A Neural Network Based Optimal Wide Area Control Scheme for a Power System", Proceedings of the IEEE Industrial Applications Society, Hong Kong, vol. 1, October, pp 700 – 706.

Venayagamoorthy, G.K., Y. del Valle, S. Mohagheghi, W. Qiao, S. Ray, R.G. Harley, D.M. Falcao, G.N. Taranto, T.M.L. Assis, 2005a. "Effects of a STATCOM, a SCRC and a UPFC on the Dynamic Behavior of a 45 Bus Brazilian Power System", Proceedings of the IEEE Power Engineering Society Inaugural 2005 Conference and Exposition in Africa, Durban, South Africa, July, pp 305-312.

Venayagamoorthy, G.K., U. O. Aliyu, J. H. Chow, J. Sanchez-Gasca, 2005b. "Modal Extraction with Three Different Methods for the Nigerian Power System", *International Conference on Power Systems, Operation and Planning*, Praia, Cape Verde, May, pp 61-66.

Venayagamoorthy, G.K., R.G. Harley and D.C. Wunsch II, 2003. "Dual Heuristic Programming Excitation Neurocontrol for Generators in a Multimachine Power System", *IEEE Transactions on Industry Applications*, vol. 39, no. 2, March/April, pp 382-392.

Venayagamoorthy, G.K., R.G. Harley and D.C. Wunsch II, 2002. "Comparison of Heuristic Dynamic Programming and Dual Heuristic Programming Adaptive Critics for Neurocontrol of a Turbogenerator", *IEEE Transactions on Neural Networks*, vol. 13, no. 3, May, pp 764-773.

Venayagamoorthy, G.K. and R.G. Harley, 2001. "A Continually Online Trained Neurocontroller for Excitation and Turbine Control of a Turbogenerator", *IEEE Transactions on Energy Conversion*, vol. 16, no. 3, September, pp 261 - 269.

Wang, H.F., 2001. "Multi-Agent Coordination for the Secondary Voltage Control in Power System Contingencies", *IEE Proceedings- Generation, Transmission, Distribution*, vol. 148, no. 1, January, pp 61-66.

Wang, H.F., 1999. "Philips-Heffron Model of Power Systems Installed with STATCOM and Applications", *IEE Proceedings- Generation, Transmission, Distribution*, vol. 146, no. 5, September, pp 521-527.

Welch, G. and G. Bishop, 2004. "An Introduction to the Kalman Filter", Technical Report, University of North Carolina at Chapel Hill, April, pp 1-15.

Werbos, P.J., 2000. "New Directions in ACDs: Keys to Intelligent Control and Understanding the Brain", Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, vol. 3, July, pp 61-66.

Werbos, P.J., 1999. "Tutorial on Neurocontrol, Control Theory and Related Techniques: From Backpropagation to Brain-Like Intelligent Systems", Tutorial Presented at the 12th International Conference on Mathematical and Computer Modeling and Scientific Computing (ICMCM & SC), Chicago, IL, USA, August.

Werbos, P.J., 1994. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*, Wiley, ISBN 0-471-59897-6.

Werbos, P.J., 1992. "Neurocontrol and Supervised Learning: An Overview and Evaluation", in *Handbook of Intelligent Control*, White and D.A. Sofge, Massachusetts: Van Nostrand Reinhold, ISBN 0-442-30857-4, Chapter 3, pp 65-86.

Werbos, P.J., 1990. "A Menu of Designs for Reinforcement Learning Over Time", *Neural Networks for Control*, Massachusetts: MIT Press, pp 67-95.

Werbos, P.J., 1974. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, Ph.D. Thesis, Committee on Applied Mathematics, Harvard University.

Widrow, B., N. Gupta and S. Mitra, 1973. "Punish/Reward: Learning with a Critic in Adaptive Threshold Systems", *IEEE Transactions on System, Man and Cybernetics*, vol. 3, no. 5, pp 455-465.

Yang, S.S. and C.S. Tseng, 1996. "An Orthogonal Neural Network for Function Approximation", *IEEE Transactions on Systems, Man and Cybernetics- Part B: Cybernetics*, vol. 26, no. 5, October, pp 779-784.

Yang, Z., C. Shen, L. Zhang, M.L. Crow, and S. Atcitty, 2001. "Integration of a StatCom and Battery Energy Storage", *IEEE Transactions on Power Systems*, vol. 16, no. 2, May, pp 254-260.

Yang, Z., S. Chen, M.L. Crow, and L. Zhang, 2000. "An Improved STATCOM Model for Power Flow Analysis", Proceedings of the IEEE Power Engineering Society Summer Meeting, vol. 2, July, pp 1121-1126.

Yao, Z., P. Kesimpar, V. Donescu, N. Uchevin and V. Rajagopalan, 1998. "Nonlinear Control for STATCOM Based on Differential Algebra", 29th Annual IEEE Power Electronics Conference, May, vol. 1, pp 329-334.

Ying, H., Y. Ding, S. Li and S. Shao, 1999. "Comparison of Necesssary Conditons for Typical Takagi-Sugeno and Mamdani Fuzzy Systems as Uninersal Approximators", *IEEE Transactions on Systems, Man and Cybernetics- Part A: Systems and Humans*, vol. 29, no. 5, September, pp 508-514.

Yixin, N., M.L. On, H. Zhenyu, C. Shousun, Z. Baolin, 1999. "Fuzzy Logic Damping Controller for FACTS Devices in Interconnected Power Systems", Proceedings of the IEEE International Symposium on Circuits and Systems, vol. 5, May, pp 591-594.

Yoo, T.S. and S. Lafortune, 2004. "Decentralized Supervisory Control with Conditional Decisions: Supervisor Existence", *IEEE Transactions on Automatic Control*, vol. 49, no. 11, November, pp 1886-1904.

Zadeh, L.A., 1965. "Fuzzy Sets", *Information and Control*, vol. 8, pp 338-353.

Zoltawski, M., 1988. "Novel Techniques for the Estimation of Array Signal Parameters Based on Matrix Pencil, Subspace Rotations, and Total Least Squares", Proceedings of *IEEE ICASSP,* vol. 4, April, pp. 2861-2864.

VITA

Salman Mohagheghi was born in Manchester, UK in 1976. He completed his B.Sc. and M.S degrees in Electrical Power Engineering from University of Tehran, Tehran, Iran and Sharif University of Technology, Tehran, Iran in 1998 and 2001 respectively. He joined Georgia Institute of Technology, Atlanta GA in August 2001, where he has been working towards his PhD degree.

Since he started his studies at Georgia Tech, Salman has been a graduate research assistant working on power systems operation and control, nonlinear control systems and computational intelligence. He has also been teaching/supervising various laboratories and courses on power electronics, electric machines and energy conversion systems, analog/digital circuit design and applications of computational intelligence in power systems.

In addition to his work related activities, Salman is an active member of the Georgia Tech family. He was the President of the Iranian Student Association in 2003-05 and the Vice-President of the IEEE Power Engineering Society Student Chapter at Georgia Tech in 2003-05.